

# Словарь программиста



## Русский язык

---

**Алгоритм** – порядок действий, которые необходимо выполнить для решения определенной задачи. В программировании алгоритмы описываются средствами псевдокода, блок-схем и UML диаграмм.

**А** **Ассемблер (Assembler)** – язык программирования низкого уровня, инструкции которого соответствуют инструкциям машинного кода. Также, ассемблером называют программу – транслятор с языка программирования низкого уровня в *машинный код*.

---

**Б** **Блок-схема** – графическая нотация для описания алгоритмов. Используется программистами в процессе разработки и анализа логики работы программных компонентов.

---

**Веб-дизайн** – проектирование и разработка внешнего вида интернет сайтов и пользовательского интерфейса веб-приложений, в том числе художественное оформление в соответствии с определенной стилистикой (корпоративным стилем компании-заказчика, стилем рекламной кампании и т.д.). Веб-дизайн является частью веб-программирования.

**В** **Веб-программирование** – направление в программировании, ориентированное на разработку приложений для сети интернет (веб-приложений). Пользователь взаимодействует с веб-приложением через интернет браузер. Само приложение загружается с удаленного веб-сервера, а общение клиента и сервера осуществляется через HTTP протокол.

---

**Веб-сервис, веб-служба (Web service)** – вид программного обеспечения, предоставляющего свою функциональность клиентам на уровне программных интерфейсов, построенных на базе HTTP протокола. Веб-сервис, как и любой другой ресурс всемирной паутины идентифицируется веб-адресом (URI, Uniform Resource Identifier, унифицированный идентификатор ресурсов). Веб-сервис - это частный случай веб-приложения, отличающегося тем, что не имеет пользовательского интерфейса. Клиентом веб-сервиса обычно является основное веб-приложение или другой веб-сервис, поскольку сам по себе веб-сервис - это компонент сервис-ориентированной архитектуры распределенного приложения.

**Веб-сайт, интернет-сайт (Web site)** – основной вид ресурсов всемирной паутины, представляющей собой совокупность веб-страниц, созданных на основе языка разметки HTML и объединенных общим доменным именем. Сайт размещается на веб-сервере, который является узлом глобальной компьютерной сети интернет. Доступ к сайту осуществляется средствами интернет браузера через HTTP протокол. Веб-сайт является частным случаем веб-приложения.

**Венгерская нотация** – набор соглашений и правил об именовании переменных, констант, типов, процедур, функций и других объектов программ с использованием суффиксов и префиксов, отражающих тип, область видимости и прочие их аспекты.

---

**Геоинформационная система (ГИС)** – информационная система, в задачи которой также входит хранение, графическое отображение и управление доступом к пространственным (географическим) данным.

**Графический пользовательский интерфейс** - см. GUI.

---

**Д Делфи** - см. Delphi.

---

**И Inspeкция кода (Code review)** – систематический и периодический анализ программного кода, направленный на поиск обнаруженных на ранних стадиях разработки программного продукта ошибок, а также, на выявление некачественных архитектурных решений и критических мест в программе.

**Информационная система** – совокупность данных и обслуживающих эти данные технических, программных и организационных ресурсов, создаваемая с целью информационной поддержки пользователей.

---

**К Кастомизация** в контексте разработки программных продуктов может

---

означать процесс настройки функциональности программного продукта под требования конечного потребителя. В процессе кастомизации программных продуктов, создающихся на базе некоторого каркаса, зачастую пишутся процедуры на встроенных языках бизнес - сценариев или разрабатываются внешние программные модули, встраиваемые в общий каркас по определенным правилам – спецификациям Plug-In компонентов. Таким образом, кастомизация приводит к расширению базовой функциональности стандартных программных решений усилиями организаций, занимающихся их внедрением, а разработчики основного программного продукта лишь предоставляют набор инструментов для этого.

**Кристофер Дейт (Christopher J. Date)** – специалист и очень известный теоретик в области реляционных баз данных, по научным работам которого преподаются основы баз данных во многих учебных заведениях мира.

---

**Облачные вычисления (Cloud computing)** – модель организации вычислений, при которой вычислительные процессы, запрашиваемые клиентом, происходят на удаленных, намного более мощных по сравнению с клиентскими вычислительных ресурсах. Взаимодействие сервера с клиентом осуществляется посредством сетевого доступа, но сам процесс вычислений для пользователя является неразрывным – как будто все происходит на стороне клиента.

**Объектно-ориентированная база данных** – база данных, основанная на *объектно-ориентированной модели данных*. Концепции объектно-ориентированного программирования и объектно-ориентированного проектирования баз данных основываются на общих подходах к моделированию предметной области. Главными элементами объектно-ориентированной модели являются классы, объекты, интерфейсы, атрибуты (свойства), методы и т.п.

**Объектно-ориентированное программирование (ООП)** – самая удачная и гениальная, на мой взгляд, парадигма программирования. Суть объектно-ориентированного программирования в представлении обрабатываемой информации в виде объектов – экземпляров классов. Класс – это новый (по отношению к процедурному программированию) тип данных, который объединяет в себе и структуры данных и параметризованные процедуры.

---

**Паскаль** - см. Pascal.

**II**

**Прикладное программирование** – процесс разработки программного

---

обеспечения, предназначенного для решения прикладных задач в определенной сфере деятельности. Такое программное обеспечение называют прикладным, и оно характеризуется тем, что не использует вычислительные ресурсы аппаратного обеспечения напрямую, а делает это посредством операционной системы.

---

**Регулярные выражения** – язык шаблонов или язык масок для поиска в тексте фрагментов, удовлетворяющих определенному набору критериев, разбиения найденных фрагментов на группы с целью дальнейшего их анализа и обработки.

**Реляционная база данных** — база данных, основанная на реляционной модели данных.

**Р**

**Рефакторинг кода (Refactoring)** – процесс внесения изменений в программный код в соответствии с некоторым набором правил – приемов рефакторинга, которые не меняют смысл программы, но делают ее код более стройным и легким для интерпретации человеком. Также, рефакторинг способствует облегчению поиска ошибок и “узких мест” в программе.

---

**Сервис - Ориентированная Архитектура (COA)** – архитектура сложных программных комплексов, состоящих из отдельно функционирующих программных компонентов – модулей системы, взаимодействующих между собой по сетевым протоколам. Подход COA основан на классификации и унификации типовых модулей системы и предоставляемых ими интерфейсов, а также, на возможности замены одних модулей их аналогами без ущерба функционированию всей системы. Программные модули характеризуются возможностью многократного использования, автономностью, слабой связностью и чаще всего реализуются, как отдельные веб-сервисы.

**С**

**Система Управления Базами Данных (СУБД)** – программное обеспечение (информационная система), осуществляющее весь спектр операций по управлению базами данных, к которым относятся сама организация хранения данных, обработка инструкций SQL, организация резервного копирования, восстановление резервных копий и т.п.

**Система Управления Содержимым (Content Management System, CMS)** – программное обеспечение (информационная система), предоставляющее возможности управления содержимым (контентом) целевой информационной системы или сайта и являющееся фреймворком по отношению к нему. Одной из функций

---

управления содержимым является предоставление на базе ролей или профилей пользователей доступа к этому самому содержимому.

---

**У** **Управляемый код (Managed code)** – программный код, исполняемый виртуальной машиной, например, CLR.

---

**Ф** **Фреймворк (Framework)** – вид программного обеспечения, являющегося основой (каркасом) различных прикладных программных продуктов. Фреймворк упрощает процесс функционального объединения программных компонентов различного назначения, их взаимодействия и настройки, что позволяет быстро и качественно собрать целевой программный продукт, удовлетворяющий требованиям конкретного потребителя.

---

## English

---

**ActiveX Data Objects (ADO)** – очередное развитие сервисов унифицированного доступа к базам данных от Microsoft на базе технологии *ActiveX*, которая, в свою очередь, расширяет возможности COM компонентов.

**ActiveX Data Objects.NET (ADO.NET)** – последнее поколение программной технологии унифицированного доступа к базам данных от компании Microsoft, которое уже основано на платформе .NET Framework. Нужно отметить, что технология .NET Framework лишь воплощает развитие идей COM и *ActiveX*, но не следующий уровень их реализации, поскольку является абсолютно новой программной платформой.

**Application programming interface (API)** – *интерфейс программирования приложения* – функциональность приложения, доступная на программном уровне внешним программным компонентам. API, чаще всего, представляет собой набор программных компонентов, интерфейсов, глобальных функций и т.п. API приложения используется для интеграции с другими приложениями или для расширения и настройки собственной функциональности.

**Assembler** - см. Ассемблер.

---

**В** **Base Class Library (BCL)** – Общая для всех языков программирования платформы Microsoft .NET Framework библиотека классов, в состав которой входят компоненты работы с различными структурами данных, файловыми структурами, базами данных, ресурсами в интернете и т.д.

---

**С** **Common Intermediate Language (CIL)** – промежуточный язык программирования низкого уровня, обрабатываемый CLR. По мере обработки *CIL* кода, виртуальная машина CLR переводит его в

---

машинный - *платформо-зависимый код*. *CIL код* является результатом предварительной компиляции программ, написанных на языках программирования высокого уровня платформы Microsoft .NET Framework, таких как C# и VB.NET.

**Common Language Runtime (CLR)** – исполняющая среда программной платформы Microsoft .NET Framework, интерпретирующая и контролирующая выполнение программного кода, представленного промежуточным CIL кодом.

**Common Language Specification (CLS)** – спецификация, которая включает в себя набор требований к *CLS совместимому языку программирования*, выполнение которых гарантирует возможность использования разработанных на нем программных компонентов из программ, написанных на других CSL совместимых языках. Речь идет о языках программирования, создаваемых для программной платформы Microsoft .NET Framework, которые заведомо могут предоставлять различные возможности программистам. CLS определяет некое подмножество языков программирования для .NET Framework, в рамках которого все они обладают более или менее одинаковыми возможностями, а значит, могут совместно использоваться при написании многокомпонентных программных продуктов.

**Component Object Model (COM)** – технология разработки программных компонентов, предназначенных для повторного использования в различных приложениях. Унификация доступа к программным сервисам реализована на основе интерфейсов. Технологией COM компания Microsoft реализует идеи объектно-ориентированного проектирования программных компонентов, возможность использования которых не зависит от выбранного языка программирования и программной платформы.

**Code review** - см. Инспекция кода.

**C#, C Sharp (Си Шарп)** – объектно-ориентированный язык программирования платформы Microsoft .NET Framework. По синтаксису очень схож с языками программирования C++ и Java.

**C++** - объектно-ориентированный язык программирования, разработанный Бьерном Страуструпом.

---

**Delphi** – объектно-ориентированный язык программирования, созданный на основе языка программирования Pascal и среда разработки программных продуктов компании *Borland*.

---

**Document Object Model (DOM)** – программный интерфейс доступа к иерархическим структурам в оперативной памяти, соответствующим элементам документов, созданных на базе языков разметки, например, HTML, XML, XHTML и т.п. Обработка документов на основе DOM характеризуется полной загрузкой всего содержимого документа в оперативную память и противопоставляется обработке на основе SAX технологий. Тем не менее, DOM зачастую обеспечивает не только доступ, но и возможности программного редактирования документа.

**Document Type Definition (DTD)** – специальный язык, который, также как и XML Schema предназначен для описания синтаксических правил XML расширения. Язык DTD, в отличие от XML Schema, сам не является XML расширением, и в настоящее время считается устаревшим стандартом.

---

**ER диаграммы** – графическая нотация визуализации данных ER модели. **ER модель (модель Сущность-Связь, Entity Relationship Model)** – модель данных предметной области, используемая для проектирования реляционных баз данных в терминах сущностей и связей между ними. Сущности характеризуются именем и набором атрибутов, а связи - типом (один – к – одному, один – ко – многим, многие – ко – многим) и характеристикой {обязательная | необязательная}.

---

**F Framework** - см. Фреймворк.

**GIS** - см. ГИС.

**G Graphical User Interface (GUI)** – графический пользовательский интерфейс, предоставляемый современными операционными системами, такими как Windows (Microsoft), Mac OS (Apple) и т.п. GUI представлен графическими окнами, кнопками, списками и прочими элементами управления, манипуляция которыми осуществляется посредством мыши и (или) клавиатуры.

---

**HyperText Markup Language (HTML)** – язык разметки веб-страниц, предназначенный для форматирования их содержимого (контента), отображаемого интернет - браузерами.

**H HTTP (HyperText Transfer Protocol)** – сетевой протокол прикладного уровня на базе TCP/IP, предназначенный для передачи гипертекста в условиях клиент-серверного окружения. В основном используется для обмена данными между пользователями и ресурсами в сети интернет – является средством общения распределенных компонентов всемирной

---

паутины.

---

**Java** - объектно-ориентированный язык программирования, разработанных компанией Sun Microsystems.

**JavaScript** – встроенный в интернет браузер процедурный язык программирования. JavaScript не является чьей-либо собственностью. Назначение JavaScript – манипуляция элементами HTML разметки посредством программных интерфейсов объектной модели загруженной интернет страницы - DOM.

**jQuery** – библиотека, написанная на JavaScript и созданная с целью упрощения взаимодействия с элементами HTML разметки при создании динамических интернет страниц. jQuery предоставляет программный интерфейс запросов к данным объектной модели загруженной страницы с возможностью выполнения ряда операций над результатами этих запросов.

**Ж**

**Just-in-time compiler (JIT compiler)** - компилятор промежуточного кода, такого как *байт-код* (промежуточный язык программирования, обрабатываемый виртуальной машиной Java - JVM) или CIL код в машинный код. *JIT компиляция* происходит во время исполнения программы виртуальной машиной и затрагивает не весь код, а только тот, который необходимо исполнить в настоящий момент. Использование промежуточного кода и исполняемой его виртуальной машины дает независимость от программной платформы (операционной системы), а частичная JIT компиляция – экономию по времени в сравнении с компиляцией всего кода или в сравнении с процессом трансляции (построчной компиляции).

---

**Language Integrated Query (LINQ)** – расширение объектно-ориентированных языков программирования платформы Microsoft .NET Framework синтаксисом структурированных запросов, разработанное той же компанией. Используя LINQ, можно производить выборку набора объектов из некоторого множества (коллекции объектов) практически так же, как осуществляется отбор записей из таблиц реляционной базы данных посредством SQL, синтаксис которого во многом позаимствован авторами LINQ.

**Л**

---

**Managed code** - см. Управляемый код.

**Microsoft Developer Network (MSDN)** – подразделение компании Microsoft и созданный этим подразделением информационный сервис, предназначенный для информационной поддержки разработчиков программного и аппаратного обеспечения, использующих или

---

ориентирующихся

на *технологии*

*Microsoft.*

**Microsoft Visual Studio** – интегрированная среда разработки программных продуктов компании Microsoft, которая, в том числе, поддерживает языки программирования для платформы Microsoft .NET Framework.

**Microsoft .NET Framework** – одна из последних программных технологий компании Microsoft, созданная для разработки платформонезависимых приложений, исполняемых виртуальной машиной CLR.

---

**Open Database Connectivity (ODBC)** – API для унифицированного доступа к базам данных - предшественник OleDB.

**O Object Linking and Embedding, Database (OLE DB)** – библиотека COM интерфейсов унифицированного доступа к базам данных. OLE DB разработана Microsoft на смену ODBC.

---

**Pascal** – один из наиболее известных языков программирования высокого уровня, который широко используется в целях обучения программированию. *Объектный Паскаль (Object Pascal)* является объектно-ориентированным расширением Паскаля, на основе которого создан язык программирования Delphi.

**Plug-In** – внешний по отношению к основному программному продукту компонент, который в ходе работы основного приложения может быть динамически к нему подключен и запущен. Как правило, такие модули используют базовую функциональность программы - контейнера для ее расширения и настройки - кастомизации. Plug-In компоненты разрабатываются по определенным контейнером программным спецификациям, а доступ к его базовой функциональности получают в процессе подключения через внутренний API. Частным случаем Plug-In компонента можно считать различного рода процедуры, написанные на встроенных в контейнеры языках программирования и исполняемые также встроенными компиляторами или трансляторами. Примерами таких контейнеров являются базовые продукты различных решений компаний 1С, SAP и т.п.

**Procedural Language, Structured Query Language (PL-SQL)** – процедурное расширение языка SQL, разработанной компанией Oracle для своих СУБД.

---

**R Refactoring** - см. Рефакторинг кода

---

**Service-Oriented Architecture (SOA)** - см. Сервис-ориентированная архитектура

**Simple API for XML (SAX)** – программный интерфейс последовательного доступа к содержимому XML документа в процессе чтения XML файла или последовательного формирования XML документа в процессе записи XML файла. Преимущества, в сравнении с DOM – это экономия оперативной памяти в процессе обработки.

**S Structured Query Language (SQL)** – язык описания структурированных запросов к базам данных, результатом выполнения которых может быть или структурированный набор информации (результат выборки или кортеж) или изменения в составе данных (модификация данных). Также, инструкции SQL позволяют изменять саму структуру базы данных (определение данных), настраивать параметры доступа к различным ее разделам (определение доступа) и управлять транзакциями: применять или откатывать текущие изменения.

---

**Transact-SQL (T-SQL)** – процедурное расширение языка SQL, разработанной компанией Microsoft для своей СУБД SQL Server.

---

**Unified Modeling Language (UML)** – унифицированный язык моделирования предметной области, представленный набором нотаций к графическому представлению диаграмм классов, компонентов, прецедентов использования подсистем, бизнес-процессов и т.д. и т.п.

---

**VB.NET** – объектно-ориентированный язык программирования платформы Microsoft .NET Framework. Синтаксис во многом заимствован у другого языка программирования компании Microsoft – Visual Basic. За внешним сходством скрываются фундаментальные отличия используемых языками программных платформ. Visual Basic создавался с целью упрощения работы с программными компонентами, разрабатываемыми по технологии COM. Язык программирования VB.NET работает на той же платформе, что и язык программирования C#.

---

**Web-design** - см. Веб-дизайн.

**Web-programming** - см. Веб-программирование.

**Web-service** - см. Веб-сервис.

**W Web-site** - см. Веб-сайт.

**World Wide Web Consortium (W3C)** - международная организация, основной деятельностью которой является разработка технологических стандартов для всемирной паутины и рекомендаций по их внедрению для

---

разработчиков программного обеспечения.

---

**eXtensible HyperText Markup Language (XHTML)** – расширение XML, реализующее и расширяющее возможности HTML.

**eXtensible Markup Language (XML)** – обобщенный язык разметки, предназначенный для создания на основе себя расширений: специализированных языков разметки или текстовых форматов для хранения и обмена наборами различного рода структурированной информации.

Прочитать про XML, его применение и способы обработки...

**X eXtensible Stylesheet Language (XSL)** – расширение XML, представляющее собой язык описания трансформации XML документов в другие форматы (RTF, PDF) или XML документы, соответствующие другим его расширениям (XHTML, WML). Для составления качественно XSL документа необходимо знать XML схемы трансформируемого и целевого документов.

**XML Path Language (XPath)** – Язык структурированных запросов к элементам XML. Выполняет те же функции по отношению к XML документу, что и SQL по отношению к базе данных.

**XML Schema (Схема XML документа)** – расширение XML, предназначенное для описания правил и проверки на соответствие этим правилам *специализированных XML* документов.

---

1 **Программа** – данные, предназначенные для управления конкретными компонентами системы обработки информации в целях реализации определенного алгоритма

2 **Программное обеспечение** – совокупность программ системы обработки информации и программных документов, необходимых для эксплуатации этих программ

3 **Программирование** – научная и практическая деятельность по созданию программ

4 **Системная программа** – программа, предназначенная для поддержания работоспособности системы обработки информации или повышения эффективности ее использования в процессе выполнения прикладных программ

5 **Управляющая программа** – системная программа, реализующая набор функций управления, в который включают управление

ресурсами и взаимодействием с внешней средой системы обработки информации, восстановление работы системы после появления неисправностей в технических средствах

6 **Прикладная программа** – программа, предназначенная для решения задачи или класса задач в определенной области применения системы обработки информации

7 **Подпрограмма** – программа, являющаяся частью другой программы и удовлетворяющая требованиям языка программирования

8 **Программный модуль** – программа или функционально завершенный фрагмент программы, предназначенный для хранения, трансляции, объединения с другими программными модулями для загрузки в оперативную память

9 **Исходный модуль** – программный модуль на исходном языке программирования, обрабатываемый транслятором

10 **Объектный модуль** – программный модуль, получаемый в результате компиляции исходного модуля

11 **Загрузочный модуль** – программный модуль, представленный в форме, пригодной для загрузки в основную память для выполнения

12 **Рекурсивная подпрограмма** – подпрограмма, которая может обращаться к себе самой

13 **Система программирования** – система, образуемая языком программирования, компиляторами или интерпретаторами программ, представленных на данном языке, соответствующей документацией, а также вспомогательными средствами для подготовки программ к форме, пригодной для выполнения

14 **Язык высокого уровня** – язык программирования, понятия и структура которого удобны для восприятия человеком

15 **Транслятор** – программа или техническое средство, выполняющие трансляцию программы.

*Примечание: на транслятор обычно возлагаются функции диагностики ошибок, формирования словарей идентификаторов, выдачи для печати текстов программ и т. д.*

16 **Конвектор языка** – транслятор с некоторого языка на другой язык такого же уровня

17 **Компилятор** – программа или техническое средство, выполняющее компиляцию

18 **Интерпретатор** – программа или техническое средство, выполняющее интерпретацию

19 **Структурное программирование** – метод построения программ, использующий только иерархически вложенные конструкции, каждая из которых имеет единственную точку входа и единственную точку выхода

20 **Объектно-ориентированное программирование** – метод построения программ как совокупностей объектов и классов объектов, которые могут вызывать друг друга

*Примечание: объекты состоят из данных и операций над данными*

21 **Логическое программирование** – метод построения программ как совокупности логических правил с предварительно определенными алгоритмами для обработки входных данных программы в соответствии с ее правилами

22 **Спецификация программы** – формализованное представление требований, предъявляемых к программе, которые должны быть удовлетворены при ее разработке, а также описание задачи, условия и эффекта действия без указания способа его достижения

23 **Трансляция программы** – преобразование программы, представленной на одном языке программирования, в программу на другом языке и в определенном смысле равносильную первой

24 **Компиляция** – трансляция программы с языка высокого уровня в форму, близкую к программе на машинном языке

25 **Ассемблирование** – компиляция программ с языка ассемблера

26 **Поиск ошибок (в программе)** – деятельность, в результате которой выявляются ошибки в программе с целью их последующего исправления

27 **Верификация программы** – доказательство того, что работа программы соответствует спецификации на эту программу

28 **Дамп** – данные, представляющие собой полное или частичное содержимое оперативной памяти, выводимое на периферийное устройство

29 **Аварийны дампы** – дампы, полученный в результате ненормального завершения программы

30 **Тупиковая ситуация** – ситуация, в которую попадают две или несколько асинхронных процедур, характеризующаяся невозможностью дальнейшего выполнения из-за взаимных зависимостей

31 **Функция адресации** – функция, реализуемая определенными компонентами системы обработки информации, сопоставленная с пространством памяти, определенная на множестве адресов в этом пространстве памяти и предназначенная для выделения по адресу единственной подобласти в этом пространстве памяти так, что обеспечивается определенная операция с этой подобластью

32 **Адрес в пространстве памяти** – элемент множества порций данных, являющегося областью определения функций адресации

33 **Пространство памяти** – область памяти, некоторая совокупность подобластей которой является областью значений функции адресации

34 **Указатель области памяти** – адрес области памяти, размещенный в пространстве памяти, в котором расположена эта область

35 **Адрес команды** – адрес области памяти, которая занята командой

36 **Исполнительный адрес** – адрес операнда команды, содержащийся в ней или вычисляемый на основании содержимого его полей

37 **Базовый адрес** – аддитивная часть исполнительного адреса, постоянная для определенной совокупности адресуемых данных

38 **Индекс адреса** – аддитивная часть адреса, предназначенная для модификации исполнительного адреса на отдельно вычисляемое значение в целях обращения к порциям данных, размещенным в памяти по определенному закону

39 **Базовая адресация** – способ вычисления исполнительного адреса, при котором он вычисляется как сумма базового адреса и смещения

40 **Индексирование адреса** – способ вычисления исполнительного адреса, при котором учитывают индекс адреса

41 **Цикл** – последовательность команд в программе, которая должна исполняться неоднократно в результате перехода от начала последовательности к концу

42 **Флажок (в программе)** – переменная, регистрирующая появление определенного события или состояния

43 **Переключатель** – управляемый флажком выбор одного перехода из группы возможных переходов в программе

44 **Семафор** – переменная, используемая для обеспечения взаимного исключения

45 **Общая переменная** – переменная, доступная двум или большему количеству асинхронных процедур или одновременно исполняемых программ

46 **Порция данных** – данные, представленные как целое в конкретном контексте их описания или обработки и неразрывно связанные со своим носителем

*Примечание: контексты существенно зависят от решаемых задач и этапов их решения и могут изменяться от задачи к задаче и от одного этапа к другому*

47 **Литерная цепочка** – порция данных, состоящая из последовательности литер

48 **Идентификатор** – литерная цепочка, выступающая в определенном контексте в роли символа

*Примечание: в обработке данных рассматривают идентификаторы порций данных, областей памяти, пространств памяти, томов и др., при этом они могут выражать определенные свойства этих объектов*

49 **Составной идентификатор** – идентификатор объекта, включающий идентификаторы классов, которые вложены друг в друга

50 **Область памяти** – память, выделенная для размещения одной или нескольких порций данных

51 **Подобласть памяти** – область памяти, рассматриваемая как составная часть другой области памяти

52 **Буфер** – рабочая область памяти при пересылке данных

*Примечание: при операции ввода данные заносят в буферную область*

**53 Поле данных** – неразрывная область памяти, имеющая определенное назначение и обычно снабженная именем или идентификатором

**54 Экстент памяти** – неразрывная область в пространстве внешней памяти данных, используемая при его распределении

*Примечание: в ОС ЕС ЭВМ под набор данных на устройствах прямого доступа пространство памяти отводится экстендами*

**55 Процесс обработки данных** – система действий, реализующая определенную функцию в системе обработки информации и оформленная так, что управляющая программа данной системы может перераспределять ресурсы этой системы в целях обеспечения мультипрограммирования.

*Примечания:*

1) *Процесс характеризуется состояниями, которые определяются наличием тех или иных ресурсов в распоряжении процесса и, следовательно, возможностью выполнять действия, относящиеся к процессу.*

2) *Перераспределение ресурсов, выполняемое управляющей программой, влияет на продолжительность процесса обработки данных, но не на его конечный результат.*

3) *Процесс оформляют с помощью специальных структур, которыми манипулирует управляющий механизм.*

**56 Параллельные процессы** – процессы обработки данных, у которых интервалы времени выполнения перекрываются за счет использования различных ресурсов одной и той же системы

**57 Конкурирующие процессы** – процессы обработки данных, выполняющиеся в течение определенного интервала времени с использованием одного и того же набора ресурсов путем их перераспределения

**58 Системный процесс** – процесс обработки данных, в виде которого функционируют определенные компоненты управляющей программы

**59 Процесс системного ввода** – системный процесс, предназначенный для автоматического ввода в систему обработки информации пакета заданий через назначенное этому процессу устройство ввода-вывода

*Примечание: процесс системного ввода в некоторых системах обработки информации интерпретирует предложения языка управления заданиями*

**60 Процесс системного вывода** – системный процесс, предназначенный для автоматического вывода определенной части результатов выполнения заданий наряду с сообщениями управляющей программы через назначенное этому процессу устройство ввода-вывода

**61 Приоритет процесса** – преимущественное право процесса обработки данных перед другими процессами на использование ресурсов, выражаемое числом, присвоенным этому процессу

62 **Мультипрограммная связь** – совокупность процессов, выполнением которых управляет одна и та же управляющая программа

63 **Удаленный ввод заданий** – ввод заданий вычислительной системе средствами телеобработки данных

64 **Диалоговый удаленный ввод заданий** – удаленный ввод заданий, при котором ввод осуществляется в диалоговом режиме

65 **Сеанс работы** – интервал времени, в течение которого пользователю выделены ресурсы системы обработки информации для выполнения его работ и связи с ним в интерактивном режиме

66 **Сообщение системы** – сообщение любой из системных программ, входящих в операционную систему

67 **Команда оператора** – обращение оператора системы обработки информации к управляющей программе, переданное посредством консоли, для оказания воздействия на ход обработки данных

68 **Интерпретация** – реализация смысла некоторого синтаксически законченного текста, представленного на конкретном языке

69 **Процедура начальной загрузки** – ввод программы в оперативную память для выполнения, при котором никакие программы, кроме вводимой, не управляют вводом

*Примечание: присутствие других программ в оперативной памяти не предполагается*

70 **Инициализация** – приведение областей памяти в состояние, исходное для последующей обработки или размещения данных

71 **Инициирование** – создание условий для запуска процесса обработки данных

72 **Искусственный язык** – язык, созданный отдельным лицом или группой лиц с определённой целью

73 **Программный документ** – документ, содержащий в зависимости от назначения данные, необходимые для разработки, производства, эксплуатации и сопровождения программы или программного средства

*Примечание: программные документы и их части могут выполняться любым способом на любом материале, вводиться в ЭВМ и передаваться по каналам связи*

74 **Кодирование данных** – процесс построения данных из элементов конечного множества по установленным правилам

75 **Код данных** – система, образуемая кодовым набором и правилами, по которым из элементов этого кодового набора строят данные при кодировании

76 **Кодс исправлением ошибок** – код данных, в котором допустимому кодовому представлению соответствует несколько недопустимых, причем возникновение ошибок в допустимом представлении преобразует его к одному из соответствующих недопустимых представлений, что позволяет исправить возникшую ошибку

*Примечание: примером кода с исправлением ошибок является код Хемминга*

77 **Код с обнаружением ошибок** – код данных, в котором каждое кодовое представление удовлетворяет установленным критериям так, что если в представлении возникают ошибки, то оно перестает удовлетворять этим критериям и устанавливается наличие ошибки

*Примечание: примером кода с обнаружением ошибок является циклический код*

78 **Нотация** – код данных, элементами кодового набора которого являются литеры

79 **Буквенно-цифровое представление** – кодовое представление посредством букв, десятичных цифр, пробела и специальных литер

80 **Алфавит** – кодовый набор, в котором установлено отношение порядка

81 **Кодон** – элемент кодового набора

82 **Управляющий кодон** – кодон, определенный правилами кодирования данных как оказывающий воздействие на их запись, преобразование, передачу и истолкование

83 **Литера** – кодон, отличающийся от управляющего кодона

84 **Специальная литера** – литера, отличающаяся от буквы, цифры и пробела

85 **Пробел** – литера, воспроизводимая отсутствием начертания

86 **Буква** – литера, по своему начертанию совпадающая с одной из элементов традиционного алфавита естественного языка

87 **Цифра** – литера, входящая в кодовый набор представления чисел и обозначающая одно из натуральных чисел

88 **Десятичная цифра** – цифра десятичной системы счисления.

*Примечание: десятичные цифры изображаются литерами 0, 1, 2, 3, 4, 5, 6, 7, 8, 9*