

Внимание! Возможны орфографические ошибки.

5. Особенности двоичной системы счисления.

Двоичная система счисления выделяется среди других тем, что она наиболее полным образом приспособлена к технической реализации средствами электроники. Ноль задается низким уровнем напряжения (можно упрощенно считать, что при этом напряжение отсутствует), единица — высоким уровнем. Эта схема была реализована еще в докомпьютерную эпоху развития вычислительной техники в виде цепей из телеграфных реле, которые имели два состояния — замкнутое и разомкнутое. Такие электромеханические счетные машины были созданы и выдавали прекрасные результаты производительности для своего времени, намного (порядка 1000 раз) обгоняя чисто механические. Теоретическое обоснование было положено диссертационной работой Клода Шеннона в 1937 г, на принципах, изложенных в этой работе, основана, по сути, вся цифровая техника. В то время, как механические счетные устройства могли успешно оперировать и десятичной системой счисления, электронные способны эффективно работать только с двоичной. К настоящему времени накоплен достаточно большой опыт разработки логических устройств на двоичной системе, говорящий о том, что отказ от нее в будущем развитии цифровой техники вряд ли произойдет.

6. Правила перехода между двоичной и десятичной системами счисления.

Переход от двоичной системы к десятичной.

1) надписываем над цифрами двоичного числа номера разрядов — числа, начинающиеся с нуля справа налево, пример:

9876543210

1100101100

2) составляем сумму из разрядных цифр числа, умноженных на соответствующие степени двойки (показатели степени равны номерам разрядов):

$$1 \cdot 2^9 + 1 \cdot 2^8 + 0 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 512 + 128 + 32 + 8 + 4 = 684$$

Переход от десятичной системы к двоичной.

1) делим число, записанное в десятичной системе, на 2 с остатком, неполное частное снова делим на 2 с остатком — и так далее до тех пор, пока неполное частное не станет равным 0;

2) выписываем остатки от деления в обратном порядке, пример:

$$35:2=17, \text{ остаток } 1;$$

$$17:2=8, \text{ остаток } 1;$$

$$8:2=4, \text{ остаток } 0;$$

$$4:2=2, \text{ остаток } 0;$$

$$2:2=1, \text{ остаток } 0;$$

$$1:2=0, \text{ остаток } 1.$$

$$\text{Итак, } 35_{10} = 100011$$

7. Арифметические действия в двоичной системе счисления.

Арифметические операции: сложение, вычитание, умножение, деление в двоичной системе счисления выполняются по обычным правилам этих действий «в столбик», примеры:

$$3+3=6$$

$$\begin{array}{r} 11 \\ + \\ 11 \\ \hline 110 \end{array}$$

Комментарий: в младшем (нулевом) разряде выполняем сложение: $1+1=10_2$, 0 пишем, 1 уходит в следующий разряд. В этом следующем (первом) разряде выполняем сложение: $1+1+1=11_2$, 1 пишем, 1 уходит в следующий (второй) разряд.

Умножение:

$$2*2=4$$

$$\begin{array}{r} 10 \\ * 10 \\ \hline 00 \\ + 10 \\ \hline 100 \end{array}$$

$$2*3=6$$

$$\begin{array}{r} 11 \\ * 10 \\ \hline 00 \\ + 11 \\ \hline 110 \end{array}$$

$$3*3=9$$

$$\begin{array}{r} 11 \\ * 11 \\ \hline 11 \\ + 11 \\ \hline 1001 \end{array}$$

Комментарий: умножение первого числа на разрядные единицы второго числа происходит по обычным правилам умножения без каких-либо отличий от десятичной системы, сложение было разобрано выше.

17. Наиболее популярные файловые системы

Файловая система — порядок, определяющий способ организации, хранения и именования данных на носителях информации в компьютерах, а также в другом электронном оборудовании: цифровых фотоаппаратах, мобильных телефонах и т. п.

Файловая система связывает носитель информации с одной стороны и правила для доступа к файлам — с другой. Когда прикладная программа обращается к файлу, она не имеет никакого представления о том, каким образом расположена информация в конкретном файле, так же, как и на каком физическом типе носителя (CD, жёстком диске, магнитной ленте, блоке флеш-памяти или другом) он записан. Всё, что знает программа — это имя файла, его размер и атрибуты. Эти данные она получает от драйвера файловой системы. Именно файловая система устанавливает, где и как будет записан файл на физическом носителе (например, жёстком диске).

Однако файловая система не обязательно напрямую связана с физическим носителем информации. Существуют виртуальные файловые системы, а также сетевые файловые системы, которые являются лишь способом доступа к файлам, находящимся на удалённом компьютере.

Основные функции любой файловой системы нацелены на решение следующих задач:

- 1) именование файлов;
- 2) программный интерфейс работы с файлами для приложений;
- 3) отображения логической модели файловой системы на физическую организацию хранилища данных;
- 4) организация устойчивости файловой системы к сбоям питания, ошибкам аппаратных и программных средств;
- 5) содержание параметров файла, необходимых для правильного его взаимодействия с другими объектами системы (ядро, приложения и пр.).
- 6) защита файлов одного пользователя от несанкционированного доступа другого пользователя, обеспечение совместной работы с файлами.

По предназначению файловые системы можно классифицировать на нижеследующие категории.

1. Для носителей с произвольным доступом (например, жёсткий диск): FAT32, HPFS, ext2 и др. Для прироста скорости доступа во многих файловых системах применяется асинхронная запись изменений на диск, например, журналирование: ext3, ReiserFS, JFS, NTFS, XFS, либо механизм soft updates и др. Журналирование широко распространено в Linux, применяется в NTFS. Soft updates — в BSD системах.
2. Для носителей с последовательным доступом (например, магнитные ленты): QIC и др.

3. Для оптических носителей — CD и DVD: ISO9660, HFS, UDF и др.
4. Виртуальные файловые системы: AEFS и др.
5. Сетевые файловые системы: NFS, CIFS, SSHFS, GmailFS и др.
6. Для флэш-памяти: YAFFS, ExtremeFFS, exFAT.
7. Специализированные файловые системы: ZFS (собственно файловой системой является только часть ZFS), VMFS (т. н. кластерная файловая система, которая предназначена для хранения других файловых систем) и др.

18. Файлы, каталоги, пути, полное имя файла

Все программы и данные хранятся в долговременной (внешней) памяти компьютера в виде файлов.

Файл — это определенное количество информации (программа или данные), имеющее имя и хранящееся в долговременной (внешней) памяти.

Имя файла, как правило, состоит из двух частей, разделенных точкой: собственно имя файла и расширение, помогающее пользователю и компьютеру **определить его тип** (программа, данные и т. д.).

Некоторые операционные системы или менеджеры файлов могут устанавливать соответствия между расширениями файлов и приложениями. Когда пользователь открывает файл с зарегистрированным расширением, автоматически запускается соответствующая этому расширению программа.

Популярные расширения файлов:

Исполняемые программы: exe, bin

Текстовые файлы: doc, odt, txt, rtf

Графические файлы: jpg, bmp, gif, png, tiff

Web-страницы: htm, html

Звуковые файлы: wav, mp3, midi, ogg

Видеофайлы: avi, mpeg, wmf

Код (текст) программы на языках программирования: c, pas, cpp

Очень полезная информация о расширениях файлов есть на сайте <http://www.filetypes.ru/>

Если на диске хранятся сотни и тысячи файлов, то для удобства поиска файлы организуются в многоуровневую иерархическую файловую систему. Начальный, корневой, каталог содержит вложенные каталоги 1-го уровня, в свою очередь, в каждом из них бывают вложенные каталоги 2-го уровня и т. д., причем в каталогах всех уровней могут храниться и файлы.

Для того чтобы найти файл в иерархической файловой структуре, необходимо указать путь к файлу. В путь к файлу входят записываемые через разделитель: логическое имя диска и последовательность имен вложенных друг в друга каталогов, в последнем из которых находится данный нужный файл. Путь к файлу вместе с именем файла называют полным именем файла. Примеры полного имени файлов:

OS Windows: C:\Музыка\Пикник\Иероглиф.mp3

OS Linux: /mnt/Students/feu/788/Stroynova/Таблицы/magazin.ods

Примечание: в некоторых учебных изданиях утверждается, что файловая система распознает тип файла по расширению его имени — это ошибочное утверждение надо иметь в виду при

работе с тестами, справочниками и т. д.

21. Проблема фрагментации жесткого диска.

Кластер — в некоторых типах файловых систем логическая единица хранения данных в таблице размещения файлов, объединяющая группу секторов. Как правило, это наименьшее место на диске, которое может быть выделено для хранения файла.

Понятие кластер используется в файловых системах FAT, NTFS, а так же HFS Plus. Другие файловые системы оперируют схожими понятиями (блоки в Unix, Linux).

Длинные файлы занимают несколько кластеров. Если запись производится на незаполненный диск, то кластеры, принадлежащие одному файлу, записываются подряд. Если диск переполнен, на нём может не быть цельной области, достаточной для размещения файла. Тем не менее, файл все-таки запишется, если на диске много мелких областей, суммарный размер которых достаточен для записи. В этом случае файл записывается в виде нескольких фрагментов.

Процесс разбиения файла на небольшие фрагменты при записи на диск называется фрагментацией. Если на диске много фрагментированных файлов, скорость чтения носителя уменьшается, поскольку поиск кластеров, в которых хранятся файлы, на жёстких дисках требует времени.

В некоторых файловых системах Unix последний блок файла может быть поделен на подфрагменты, в которые могут быть помещены «хвосты» других файлов. В NTFS маленькие файлы могут быть записаны в Master File Table (MFT). В файловой же системе FAT из-за примитивного алгоритма степень фрагментации постоянно растёт и требуется периодическая дефрагментация.

Для решения проблемы фрагментации используют утилиты-дефрагментаторы.

Дефрагментация — процесс обновления и оптимизации логической структуры раздела диска с целью обеспечить хранение файлов в непрерывной последовательности кластеров.

Существует и простой, но мало освещенный в литературе способ: переместить все файлы на новый, нефрагментированный (только что отформатированный) пустой раздел. Кроме того, разумно хранить часто обновляемые файлы (например, документы) отдельно от файлов, в которые не вносятся изменения (системное и прикладное ПО). Удаление пустых остатков кластеров достигается архивацией файлов.

24. Классификация текстовых редакторов.

1. Просто редакторы (редакторы кода)

Характерный пример: Notepad for Windows.

Назначение: написание коротких текстов для чтения с экрана: пояснений, заметок на память, файлов readme, написание программ .

а) достоинства: быстрота вызова, предельная простота, подсветка синтаксиса, развитый сервис для удобства программистов (в разной степени, в Notepad он отсутствует)

б) недостатки: отсутствие каких-либо функций форматирования текста: установки шрифтов, расположения текста на странице, вставки графики.

Резюме: эти редакторы не предназначены для написания документов и прочих текстов, которые распечатываются на бумаге.

Некоторые редакторы кода имеют весьма своеобразное управление, ориентированное на максимальную быстроту ввода текста. Наивысшим развитием этого направления редакторов является пакет Emacs — гордость общества разработки открытого программного обеспечения. Имеет очень высокий входной порог освоения, офисным работникам не

рекомендуется. Для программистов.

2. Текстовые процессоры. Возможна классификация по 2 направлениям: способ редактирования и применение.

1) по способу редактирования:

- а) логическое (WYSIWYM: what you see is what you mean) TeX/LaTeX, html ;
- б) визуальное (WYSIWYG: what you see is what you get) MS Word, OOWriter .

Логическое редактирование означает набор текста вперемешку с командами, указывающими компьютеру, как надо форматировать текст. Команды вводятся вручную самим пользователем с клавиатуры, они всегда находятся на виду и могут быть отредактированы так же, как обычные слова. Внешний вид набранного текста не имеет никакого отношения к тому, как этот текст будет выглядеть после компиляции текстовым процессором, и по сути представляет собой программу, набранную на специальном языке программирования. Увидеть конечный результат можно только после компиляции текстовым процессором. Всегда предполагается наличие 2 различных программ: редактора кода и компилятора. Как правило, редактор кода не имеет никакого отношения к текстовому процессору и может быть любым, каждый пользователь выбирает наиболее удобный для себя самостоятельно. Характерной чертой этого способа редактирования текста является наличие в конце работы 2 файлов: исходного текста (source code) и откомпилированного.

а) преимущества: максимальная гибкость управления, тонкая настройка с любой степенью точности, возможность определять собственные функции. Как правило, source code имеет очень маленький объем, что важно для экономии трафика при передаче по компьютерным сетям.

б) недостатки: отсутствие наглядности, высокий входной порог (требуется знание языка и навыки программирования).

WYSIWYG означает отображение на экране набираемого текста сразу в готовом виде. Команды форматирования текста создаются не пользователем, а компьютером в автоматическом режиме, они скрыты от пользователя. Если формат текста открытый, то эти команды принципиально доступны пользователю, но сложны для понимания. В случае работы в закрытом формате (MS Word) эти команды не могут быть доступны пользователю, поскольку являются коммерческой тайной фирмы Microsoft.

а) преимущества: наглядность, низкий, практически нулевой, входной порог освоения простейших базовых функций.

б) недостатки: грубое управление, ориентированное на использование мыши, проблемы с тонкой настройкой, перегруженность ненужными функциями, медленная работа как пользователя, так и программного обеспечения. Зачастую созданные WYSIWYG-редактором файлы имеют неоправданно большой объем.

2) по применению: собственно редакторы и издательские системы.

- а) редакторы: Word, Writer ;
- б) издательские системы: Tex/LaTeX, Scribus

Редакторы: предназначены для написания деловых бумаг (отчетов, заявлений, объявлений, ..) и небольших статей, содержащих большое количество таблиц и рисунков, предназначенных для публикации в журналах с низкими требованиями к качеству текста. Неплохи для создания небольших красочных плакатов и поздравительных открыток.

- а) достоинства: см. достоинства WYSIWYG ;
- б) недостатки: низкое качество текста, медленная работа, ресурсоемкость .

Издательские системы:

TeX предназначен для нужд книгоиздательства.

а) преимущества: легкая работа с огромными объемами текста (сотни и тысячи страниц одновременно), высшее (типографское) качество текста, низкая требовательность к ресурсам, также см. преимущества логического редактирования, чрезвычайная - не доступная больше ни одному текстовому процессору - возможность настройки;

б) недостатки: очень высокий входной порог. Не рекомендуется для офисных работников!

LaTeX предназначен для написания статей, в том числе большого объема, содержащих математические формулы и предназначенных для публикации в журналах с высокими требованиями к качеству текста.

а) преимущества: см. преимущества редактора TeX, кроме того: грамотная отрисовка математических формул, большое удобство при наборе и редактировании математических формул, высокое развитие средств автоматизации, при профессиональном освоении --- быстрота редактирования текста, недоступная WYSIWYG-редакторам.

б) недостатки: высокий входной порог, неудобство при работе с таблицами и рисунками.

Scribus -- WYSIWYG издательская система для Linux, в настоящее время обладает множеством недоработок, но находится в состоянии стремительного развития.

Примечание: для текстовых форматов с логическим редактированием (tex, html) существуют также графические среды редактирования позволяющие редактировать текст в режиме WYSIWYG. Характерный пример: текстовый процессор LyX для LaTeX. Созданный в результате их работы исходный текст может редактироваться и вручную при помощи редактора кода, то есть указанные форматы допускают возможность двойного редактирования: как логического, так и визуального. Зачастую графические среды имеют одновременно 2 рабочих окна: в одном отображается готовый результат с возможностью редактирования в режиме WYSIWYG, в другом отображается исходный код с возможностью логического редактирования. Компилятор работает в фоновом (невидимом для пользователя) режиме и мгновенно, "на лету", вносит изменения исходного кода в отображаемый готовый результат (напомним о быстрой работе текстовых процессоров подобного рода).

28. История возникновения Интернет.

После запуска Советским Союзом искусственного спутника Земли в 1957 году Министерство обороны США посчитало, что на случай войны Америке нужна надёжная система передачи информации. Агентство передовых оборонных исследовательских проектов США (DARPA) предложило разработать для этого компьютерную сеть. Разработка такой сети была поручена Калифорнийскому университету в Лос-Анджелесе, Стэнфордскому исследовательскому центру, Университету Юты и Университету штата Калифорния в Санта-Барбаре. Компьютерная сеть была названа ARPANET (англ. Advanced Research Projects Agency Network), и в 1969 году в рамках проекта сеть объединила четыре указанных научных учреждения. Все работы финансировались Министерством обороны США. Затем сеть ARPANET начала активно расти и развиваться, её начали использовать учёные из разных областей науки.

Первый сервер ARPANET был установлен 2 сентября 1969 года в Калифорнийском университете в Лос-Анджелесе. Компьютер Honeywell DP-516 имел 24 Кб оперативной памяти .

29 октября 1969 года в 21:00 между двумя первыми узлами сети ARPANET, находящимися на

расстоянии в 640 км — в Калифорнийском университете Лос-Анджелеса (UCLA) и в Стэнфордском исследовательском институте (SRI) — провели сеанс связи. Чарли Клайн (Charley Kline) пытался выполнить удалённое подключение к компьютеру в SRI. Успешную передачу каждого введённого символа его коллега Билл Дювалль (Bill Duvall) из SRI подтверждал по телефону.

В первый раз удалось отправить всего три символа «LOG», после чего сеть перестала функционировать. LOG должно было быть словом LOGON (команда входа в систему). В рабочее состояние систему вернули уже к 22:30 и следующая попытка оказалась успешной. Именно эту дату можно считать днём рождения Интернета.

К 1971 году была разработана первая программа для отправки электронной почты по сети. Эта программа сразу стала очень популярна.

В 1973 году к сети были подключены через трансатлантический телефонный кабель первые иностранные организации из Великобритании и Норвегии, сеть стала международной.

В 1970-х годах сеть в основном использовалась для пересылки электронной почты, тогда же появились первые списки почтовой рассылки, новостные группы и доски объявлений. Однако в то время сеть ещё не могла легко взаимодействовать с другими сетями, построенными на других технических стандартах. К концу 1970-х годов начали бурно развиваться протоколы передачи данных, которые были стандартизированы в 1982—83 годах. Активную роль в разработке и стандартизации сетевых протоколов играл Джон Постел. 1 января 1983 года сеть ARPANET перешла с протокола NCP на TCP/IP, который успешно применяется до сих пор для объединения (или, как ещё говорят, «наслоения») сетей. Именно в 1983 году термин «Интернет» закрепился за сетью ARPANET.

В 1984 году была разработана система доменных имён (англ. Domain Name System, DNS).

В 1984 году у сети ARPANET появился серьёзный соперник: Национальный научный фонд США (NSF) основал обширную межуниверситетскую сеть NSFNet (англ. National Science Foundation Network), которая была составлена из более мелких сетей (включая известные тогда сети Usenet и Bitnet) и имела гораздо большую пропускную способность, чем ARPANET. К этой сети за год подключились около 10 тыс. компьютеров, звание «Интернет» начало плавно переходить к NSFNet.

В 1988 году был разработан протокол Internet Relay Chat (IRC), благодаря чему в Интернете стало возможно общение в реальном времени (чат).

В 1989 году в Европе, в стенах Европейского совета по ядерным исследованиям (фр. Conseil Européen pour la Recherche Nucléaire, CERN) родилась концепция Всемирной паутины. Её предложил знаменитый британский учёный Тим Бернерс-Ли, он же в течение двух лет разработал протокол HTTP, язык HTML и идентификаторы URI.

Соавтор Тима Бернерса-Ли по формулировке целей и задач проекта World Wide Web в CERN, бельгийский исследователь Роберт Каиллиалу (Robert Cailliau) разъяснял позднее его понимание истоков этого проекта:

История всех великих изобретений, как это давно и хорошо известно, базируется на большом числе им предшествующих. В случае Всемирной паутины (WWW) следовало бы в этом контексте, видимо, отметить по крайней мере два важнейших для успеха проекта пути развития и накопления знаний и технологий: 1) история развития систем типа

гипертекста ...; 2) Интернет-протокол, который собственно и сделал всемирную сеть компьютеров наблюдаемой реальностью.

— Из речи на открытии Европейского отделения W3 Консорциума. Париж. Ноябрь 1995.

В 1990 году сеть ARPANET прекратила своё существование, полностью проиграв конкуренцию NSFNet. В том же году было зафиксировано первое подключение к Интернету по телефонной линии (т. н. «дозвон» — англ. Dialup access).

В 1991 году Всемирная паутина стала общедоступна в Интернете, а в 1993 году появился знаменитый веб-браузер NCSA Mosaic. Всемирная паутина набирала популярность.

Можно считать что существует две ясно различимые эры в истории Web: [до браузера Mosaic] Марка Андрессена и после. Именно сочетание веб-протокола от Тима Бернерс-Ли, который обеспечивал коммуникацию, и браузера (Mosaic) от Марка Андрессена, который предоставил функционально совершенный пользовательский интерфейс, создало условия для наблюдаемого взрыва (интереса к Веб). За первые 24 месяца, истекшие после появления браузера Mosaic, Web прошел стадию от полной неизвестности (за пределами считанного числа людей внутри узкой группы ученых и специалистов лишь одного мало кому известного профиля деятельности) до полной и абсолютно везде в мире его распространенности.

— A Brief History of Cyberspace, Mark Pesce, ZDNet, 15 октября 1995.

В 1995 году NSFNet вернулась к роли исследовательской сети, маршрутизацией всего трафика Интернета теперь занимались сетевые провайдеры, а не суперкомпьютеры Национального научного фонда.

В том же 1995 году Всемирная паутина стала основным поставщиком информации в Интернете, обогнав по трафику протокол пересылки файлов FTP. Был образован Консорциум всемирной паутины (W3C). Можно сказать, что Всемирная паутина преобразила Интернет и создала его современный облик. С 1996 года Всемирная паутина почти полностью подменяет собой понятие «Интернет».

В 1990-е годы Интернет объединил в себе большинство существовавших тогда сетей (хотя некоторые, как Фидонет, остались обособленными). Объединение выглядело привлекательным благодаря отсутствию единого руководства, а также благодаря открытости технических стандартов Интернета, что делало сети независимыми от бизнеса и конкретных компаний. К 1997 году в Интернете насчитывалось уже около 10 млн компьютеров, было зарегистрировано более 1 млн доменных имён. Интернет стал очень популярным средством для обмена информацией.

В настоящее время подключиться к Интернету можно через спутники связи, радио-каналы, кабельное телевидение, телефон, сотовую связь, специальные оптоволоконные линии или электропровода. Всемирная сеть стала неотъемлемой частью жизни в развитых и развивающихся странах.

В течение пяти лет Интернет достиг аудитории свыше 50 миллионов пользователей. Другим средствам массовой информации требовалось гораздо больше времени для достижения такой популярности

29. Понятие высказывания, логические операции, выражения.

Высказывание - первичное понятие математической логики (то есть, не имеющее определения и формируемое на основе примеров). Высказывания записываются

повествовательными предложениями, некоторые высказывания могут быть переведены на специальные символьные языки. Примеры: « $2*2=4$ », «коровы умеют летать». Как видим, высказывания могут быть истинными и ложными. Предложения: «Стой!», «Кто идет?», «Стрелять буду!» высказыванием не являются. Высказывания можно разделить на элементарные и составные - выражения. Элементарные высказывания выбираются произвольным образом, из них составляются составные высказывания при помощи логических операций. Существует 5 основных логических операций: отрицание, конъюнкция, дизъюнкция, импликация и эквиваленция. Легче всего понять суть логических операций, изучая Таблицы истинности. Всякое высказывание может быть либо истинным, либо ложным. Истина и ложь называются значениями высказываний и обозначаются символами 1 и 0.

Составное высказывание, истинное при любых значениях элементарных высказываний, называется законом логики. Доказать закон можно с помощью таблиц истинности. Составное высказывание, ложное при любых значениях элем. высказываний, наз. Противоречием.

30. Таблицы истинности основных операций логики.

1) отрицание высказывания А. Обозн: \bar{A} (читается «не А»)

A	\bar{A}
1	0
0	1

2) Конъюнкция высказываний А и В (логическое «и», логическое умножение).

Обозн: $A \wedge B$

A	B	$A \wedge B$
0	0	0
0	1	0
1	0	0
1	1	1

3) дизъюнкция высказываний А и В (логическое «или», логическое сложение). Обозн:

$A \vee B$

A	B	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1

4) импликация высказываний А и В (следствие). Обозн: $A \Rightarrow B$

A	B	$A \Rightarrow B$
0	0	1
0	1	1
1	0	0
1	1	1

5) эквиваленция высказываний А и В (равносильность). Обозн: $A \Leftrightarrow B$

A	B	$A \Leftrightarrow B$
0	0	1
0	1	0
1	0	0
1	1	1