

CONHECIMENTOS BÁSICOS DE LINGUAGENS DE PROGRAMAÇÃO RELATIVOS À LÓGICA E ESTRUTURA DE PROGRAMAÇÃO.

Esse material NÃO tem como objetivo o total aprendizado quanto a arte e técnica de programação, apenas o introdutório aos conceitos de programação, visando preparar o leitor a realização de provas de concursos públicos onde esses conceitos podem ser cobrados, para pessoas não especialistas da área de programação.

1. Conhecimentos básicos de linguagens de programação relativos à Lógica e Estrutura de programação.

Programar um computador é como ensinar uma pessoa, que não saiba nada de cozinha, a assar um bolo. Numa receita típica temos duas partes, os ingredientes, e o preparo do bolo.

RECEITA BOLO DE ANIVERSÁRIO

INGREDIENTES

- 2 xícaras de açúcar
- 3 xícaras de farinha de trigo
- 4 colheres de margarina bem cheias
- 3 ovos
- 1 1/2 xícara de leite aproximadamente
- 1 colher (sopa) de fermento em pó bem cheia

MODO DE PREPARO

1. Bata as claras em neve
2. Reserve
3. Bata bem as gemas com a margarina e o açúcar
4. Acrescente o leite e farinha aos poucos sem parar de bater
5. Por último agregue as claras em neve e o fermento
6. Coloque em forma grande de furo central untada e enfarinhada
7. Asse em forno médio, pré - aquecido, por aproximadamente 40 minutos
8. Quando espetar um palito e sair limpo estará assado

Só que um “pouco” mais complexo é a arte de programar computadores.

Alguns detalhes.

Um programa de computador é escrito numa linguagem de programação, que é um subconjunto de uma língua humana, normalmente o inglês. Existem diversas linguagens de programação, cada um com suas características específicas. Tais como C, Pascal, Cobol, Java, Basic, PHP, etc.

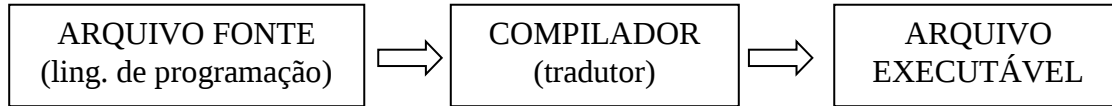
O detalhe é que um computador não entende diretamente uma linguagem de programação, quem entende essa linguagem é o ser humano, então após a escrita desse programa fonte, esse tem que ser traduzido para a linguagem do computador (o arquivo executável).

O processo de programação inicia-se com a edição de um **programa-fonte** (usando um editor de texto puro) e termina com a geração de um **programa-executável** (usando um programa “tradutor” chamado compilador).

CONHECIMENTOS BÁSICOS DE LINGUAGENS DE PROGRAMAÇÃO RELATIVOS À LÓGICA E ESTRUTURA DE PROGRAMAÇÃO.



Um



programa-fonte deve ser criado em um editor de textos (sem formatação). Após a criação, este programa é submetido a um **compilador** que analisa o código-fonte e o converte este para um **código-objeto** (ou **programa-objeto**). O código-objeto é a versão em **linguagem de máquina** do programa-fonte.

GLOSSÁRIO:

- **Código-Fonte:** criado em um editor de textos, contendo os comandos da linguagem de programação (C, Pascal...). Serve como entrada para o compilador.
 - **Programa Executável:** código que pode ser executado pelo sistema operacional. (Extensão do programa-executável: .EXE)
 - **Tempo de Compilação:** durante o processo de conversão entre código-fonte e código-objeto.
 - **Tempo de Execução:** após a ativação do programa executável.
- Podem ocorrer erros durante a compilação (o programador errou ao escrever os comandos da linguagem) ou durante a execução (o algoritmo utilizado estava errado).

Para se escrever um programa de computador temos possibilidade, independentemente da linguagem utilizada, vários comandos e estruturas de comandos. Nesse material iremos usar esses comandos com termos em português.

EXPRESSÕES RELACIONAIS E LÓGICAS

São expressões que usam os operadores relacionais e/ou lógicos. O resultado de uma expressão relacional é um valor lógico. Que pode ser VERDADEIRO ou FALSO.

Operadores Relacionais

==	Igual a
<>	Diferente
>=	Maior ou igual
>	Maior que
<	Menor que
<=	Maior ou igual

Operadores Lógicos

	OU lógico
&&	E lógico
!	Negação

Operadores Aritméticos

+	Adição
-	Subtração

**CONHECIMENTOS BÁSICOS DE
LINGUAGENS DE PROGRAMAÇÃO RELATIVOS À LÓGICA E ESTRUTURA DE PROGRAMAÇÃO.**

*	Multiplicação
/	divisão
^	Potência

Algumas linguagens podem ter outros operadores aritméticos.

Os operadores lógicos podem ser usados para ligar duas ou mais expressões lógicas.

VARIÁVEIS E TIPOS DE DADOS

A maioria das linguagens utiliza variáveis de alguns tipos como :

TEXTO OU STRING – Sequência de caracteres.

INTEIRO – somente números do conjunto dos inteiros.

REAL – somente números do conjunto dos reais.

LÓGICA OU BOOLEANA – valores lógicos VERDADEIRO ou FALSO

Algumas linguagens podem possuir outros tipos de dados ou dados estruturados.

COMANDOS DE ATRIBUIÇÃO

variavelA := valorX

A variável A passa a ter o valorX.

COMANDOS DE DECISÃO

COMANDO SE

Sintaxe:

SE (condição) FAÇA (comando1) SENÃO (comando2)

Onde : condição é uma expressão lógica, comando1 e comando2 são comandos válidos da linguagem.

O comando SE serve para alterar o fluxo de execução de um programa, também pode ser chamado de DESVIO CONDICIONAL.

COMANDO CASO

Sintaxe:

CASO OPÇÃO

valor1: comando1;

valor2: comando2;

valor3: comando3;

SENÃO

OutroComando;

CONHECIMENTOS BÁSICOS DE LINGUAGENS DE PROGRAMAÇÃO RELATIVOS À LÓGICA E ESTRUTURA DE PROGRAMAÇÃO.

O comando CASO serve para criar várias linhas alternativas do fluxo do programa dependendo do valor de uma variável enumerável. Em algumas linguagens que não proveem essa estrutura pode ser implementado por aninhamentos de SEs.

COMANDOS DE REPETIÇÃO (Loops)

Os comandos de repetição são um recurso que permite que um certo trecho do código de um programa seja **repetido** um certo número de vezes.

Existem basicamente 3 modos de repetição

ENQUANTO condição FAÇA comando;

REPETE comando ATÉ condição;

PARA variável DE início ATÉ limite FAÇA comando;

Comando ENQUANTO

O comando **ENQUANTO** permite que certo trecho de programa seja executado **ENQUANTO** certa condição for **verdadeira**

ENQUANTO <CONDIÇÃO>
FAÇA <COMANDOS>

Comando REPETE

O comando **REPETE** permite que certo trecho de programa seja executado **ATE** certa condição for **verdadeira**.

REPETE
<COMANDO>
ATÉ <CONDIÇÃO>

Comando PARA

O comando **PARA** permite que certo trecho de programa seja executado **um determinado número de vezes**.

PARA I:= 1 ATÉ 10 FAÇA
<COMANDO>

SUBPROGRAMAÇÃO

Uma função ou sub-rotina é um programa dentro de outro programa. A diferença é que uma sub-rotina executa uma série de procedimentos e volta ao programa original, uma função executa uma série de procedimentos e retorna um valor ao programa principal ao seu término.

Porque usar subprogramação?

- ✓ Para permitir o reaproveitamento de código já construído (por você ou por outros programadores);
- ✓ Para evitar que um trecho de código que seja repetido várias vezes dentro de um mesmo programa;
- ✓ Para permitir a alteração de um trecho de código de uma forma mais rápida. Com o uso de uma função é preciso alterar apenas **dentro** da função que se deseja;
- ✓ Para que os blocos do programa não fiquem grandes demais e, por consequência, mais difíceis de entender;
- ✓ Para facilitar a leitura do programa-fonte de uma forma mais fácil;
- ✓ Para separar o programa em partes (blocos) que possam ser logicamente compreendidos de forma isolada.

TERMOS

RECURSIVADE – Técnica de programação onde uma função é chamada internamente a si própria. Tem que existir uma situação em que isso não ocorra, para evitar loops infinitos.

**CONHECIMENTOS BÁSICOS DE
LINGUAGENS DE PROGRAMAÇÃO RELATIVOS À LÓGICA E ESTRUTURA DE PROGRAMAÇÃO.**

ESCOPO DE UMA VARIÁVEL – Região/tempo onde uma variável é conhecida e pode ser utilizada.

ALGORTIMOS – Uma sequência de passos para executar uma determinada tarefa.

FLUXOGRAMA – A representação gráfica de um algoritmo.

COMPILADOR – Um tipo de programa que transforma um arquivo feito em linguagem de programação, num arquivo executável diretamente pelo computador.

LINGUAGEM DE PROGRAMAÇÃO – é um método padronizado para comunicar instruções para um computador.¹ É um conjunto de regras sintáticas e semânticas usadas para definir um programa de computador. Permite que um programador especifique precisamente sobre quais dados um computador vai atuar, como estes dados serão armazenados ou transmitidos e quais ações devem ser tomadas sob várias circunstâncias. Linguagens de programação podem ser usadas para expressar algoritmos com precisão.

Uma das principais metas das linguagens de programação é permitir que programadores tenham uma maior produtividade, permitindo expressar suas intenções mais facilmente do que quando comparado com a linguagem que um computador entende nativamente (código de máquina).⁴ Assim, linguagens de programação são projetadas para adotar uma sintaxe de nível mais alto, que pode ser mais facilmente entendida por programadores humanos. Linguagens de programação são ferramentas importantes para que programadores possam escrever programas mais organizados e com maior rapidez.

Linguagens de programação também tornam os programas menos dependentes de computadores ou ambientes computacionais específicos (propriedade chamada de portabilidade). Isto acontece porque programas escritos em linguagens de programação são traduzidos para o código de máquina do computador no qual será executado em vez de ser diretamente executado.

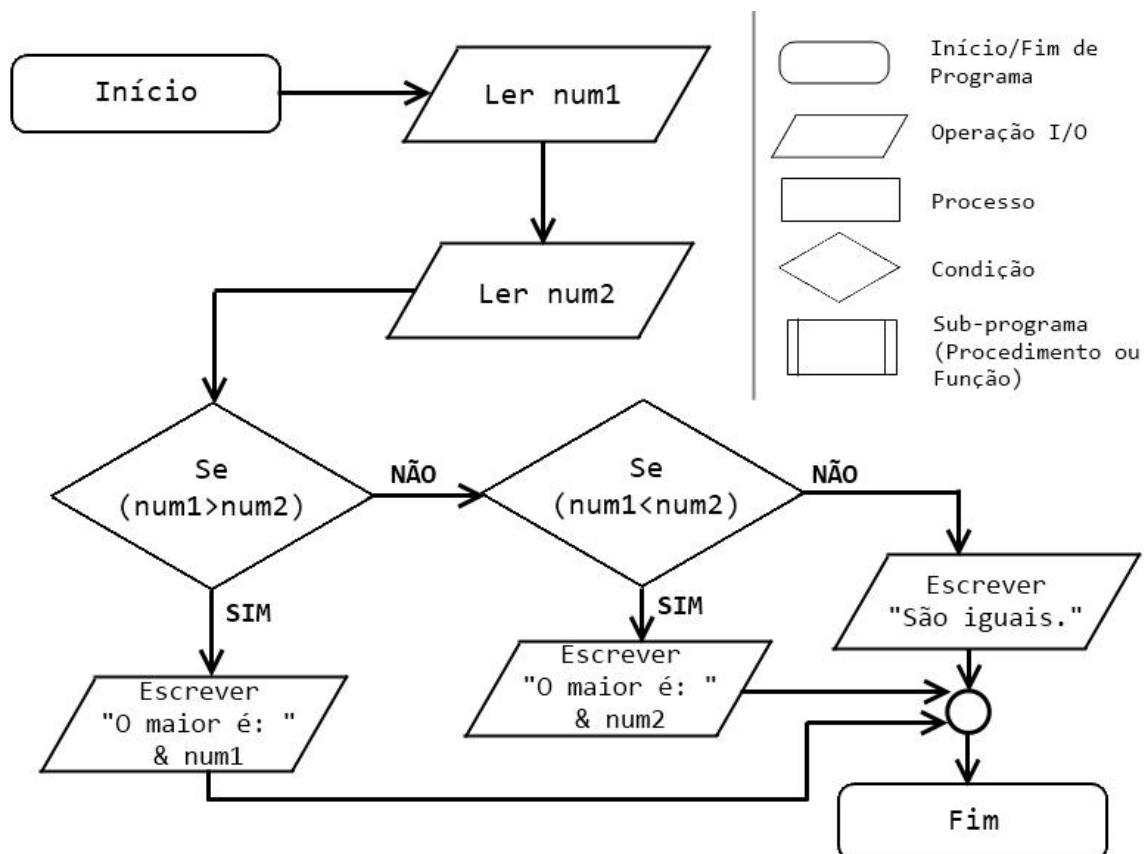
PASCAL <pre>program Hello; var mensagem : string; begin mensagem := 'Hello World!'; write(mensagem); End.</pre>	JAVA <pre>public class Main { public Main(){ System.out.println("Hello World"); } public static void main(String [] args){ Main m =new Main(); } }</pre>
C <pre>#include <stdio.h> #include <stdlib.h> int main() { printf("HELLO WORLD!!!"); return(0); }</pre>	COBOL <pre>function Hello (){ alert("Hello World!") }</pre>

**CONHECIMENTOS BÁSICOS DE
LINGUAGENS DE PROGRAMAÇÃO RELATIVOS À LÓGICA E ESTRUTURA DE PROGRAMAÇÃO.**

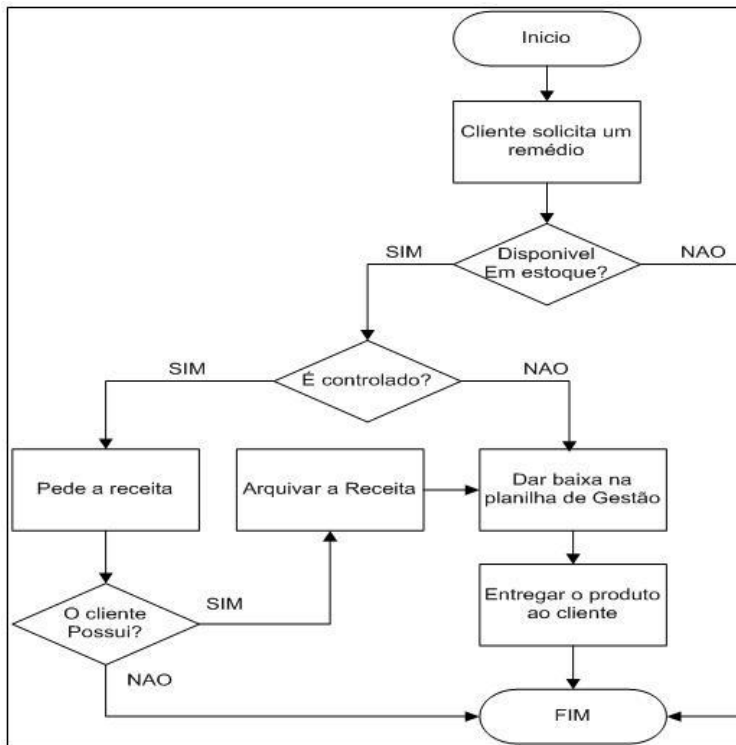
```
Sem título - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
#include<stdio.h>
main()
{
    int n,numero;
    system("cls");system("title tabuada");system("color 2b");
    printf("Qual tabuada deseja consultar?\n\n");
    scanf("%d",&numero);
    printf("\n");
    n = 1;
    while (n<=10)
    {
        printf("\t%d x %d = %d\n",numero,n,numero*n);
        n = n + 1;
    }

    printf("\n");
    putchar('\n');

    system("PAUSE");
    return(0);
}
```



**CONHECIMENTOS BÁSICOS DE
LINGUAGENS DE PROGRAMAÇÃO RELATIVOS À LÓGICA E ESTRUTURA DE PROGRAMAÇÃO.**



Algoritmo segundograu

Variaveis

A,B,C : reais

raiz1,raiz2 : reais

delta : reais

Inicio

Escreva ("Quais os valores dos coeficientes A,B e C da equação de 2º grau?")

Leia (A;B;C)

delta := $B*B-4*A*C$

Se Delta < 0

então escreva ("Não tem raizes nos reais.")

senão inicio

raiz1 := $(-1*B+raiz(Delta))/2*A$

raiz2:= $(-1*B-raiz(Delta))/2*A$

escreva (" As raizes são ";raiz1;" e "; raiz2)

fim

Fim.

**CONHECIMENTOS BÁSICOS DE
LINGUAGENS DE PROGRAMAÇÃO RELATIVOS À LÓGICA E ESTRUTURA DE PROGRAMAÇÃO.**

Algoritmo aprendendo_fatorial

Variaveis

A : reais

Função fat(N:inteiro):real

Variáveis

i : inteiro

Inicio

Se N = 0

então fat := 1

senão fat:= N * fat(n-1)

Fim

Inicio (** inicio do prog. Principal **)

Enquanto

Escreva ("Qual fatorial vc quer calcular ?, zero se quiser terminar")

Leia (A)

escreva(fat(A))

Até A = 0

Escreva ("Obrigado por me utilizar!")

Fim.

ESAF – Analise as seguintes afirmativas (Certo/Errado)

- i) O raio de atuação de uma variável qualquer é chamado de escopo.
- ii) Uma estrutura condicional representa qualquer situação em que passos devem ser repetidos até que se atinja algum objetivo.
- iii) Recursão consiste em uma função chamar a si própria com argumentos que convergem para um valor de parada, quando a recursão termina.

EASF - Na programação de computadores, um conjunto de passos que deve ser seguido para resolver um problema é denominado?

- a) Linguagem de programação
- b) Compilação
- c) Interpretador
- d) Compilador
- e) Algoritmo

CONCEITOS BÁSICOS

DE BANCO DE DADOS, DATAMINING E DATAWAREHOUSE, BANCO DE DADOS RELACIONAL

Esse material NÃO tem como objetivo o total aprendizado quanto a tecnologia existente, ao gerenciamento e administração de Banco de Dados, apenas o introdutório aos conceitos, visando preparar o leitor a realização de provas de concursos públicos onde podem ser cobrados, para pessoas não especialistas da área de banco de dados.

Definição

Bancos de dados ou **bases de dados** são coleções organizadas de informações (dados) que se relacionam de forma a criar um sentido e dar mais eficiência durante uma pesquisa ou estudo. São de vital importância para empresas, e há duas décadas se tornaram a principal peça dos sistemas de informação. Normalmente existem por vários anos sem alterações em sua estrutura.

Os programas que trabalham com os Bancos de Dados são os conhecidos como Sistemas Gerenciadores de Bancos de Dados (**SGBD**).

Existem diversos modelos de arquiteturas de Banco de Dados, mais o mais comum é o SGBD relacional.

Outros citáveis: Modelo Plano, Modelo em Rede, Modelo Hierárquico, Modelo Relacional, Orientado a objetos, e Objeto-Relacional.

- ✓ O modelo plano (ou tabular) consiste de matrizes simples, bidimensionais, compostas por elementos de dados: inteiros, números reais, etc. Este modelo plano é a base das planilhas eletrônicas.
- ✓ O modelo em rede permite que várias tabelas sejam usadas simultaneamente através do uso de apontadores (ou referências). Algumas colunas contêm apontadores para outras tabelas ao invés de dados. Assim, as tabelas são ligadas por referências, o que pode ser visto como uma rede.
- ✓ O modelo hierárquico é uma variação particular do modelo em rede, limita as relações a uma estrutura semelhante a uma árvore (hierarquia - tronco, galhos), ao invés do modelo mais geral direcionado por grafos.
- ✓ Bases de dados relacionais consistem, principalmente de três componentes: uma coleção de estruturas de dados, nomeadamente relações, ou informalmente tabelas; uma coleção dos operadores, a álgebra e o cálculo relacionais; e uma coleção de restrições da integridade, definindo o conjunto consistente de estados de base de dados e de alterações de estados. As restrições de integridade podem ser de quatro tipos: domínio (também conhecidas como type), atributo, relvar (variável relacional) e restrições de base de dados.

Assim bem diferente dos modelos hierárquico e de rede, não existem quaisquer apontadores, de acordo com o Princípio da Informação: toda informação tem de ser representada como dados; qualquer tipo de atributo representa relações entre conjuntos de dados. As bases de dados relacionais permitem aos utilizadores (incluindo programadores) escreverem consultas (*queries*) que não foram antecipadas por quem projetou a base de dados. Como resultado, bases de dados relacionais podem ser utilizadas por várias aplicações em formas que os projetistas originais não previram, o que é especialmente importante em bases de dados que podem ser utilizadas durante décadas. Isto tem tornado as bases de dados relacionais muito populares no meio empresarial.

Sistemas Gerenciadores de Bancos de dados são usados em muitas aplicações, enquanto atravessando virtualmente a gama inteira de software de computador. Os Sistemas Gerenciadores de Bancos de dados são o método preferido de armazenamento/recuperação de dados/informações para aplicações multiusuárias grandes onde a coordenação entre muitos usuários é necessária. Até mesmo usuários individuais os acham conveniente, entretanto, muitos programas de correio eletrônico e organizadores pessoais estão baseados em tecnologia de banco de dados padrão.

TRANSAÇÕES

É um conjunto de procedimentos que é executado num banco de dados, que para o usuário é visto como uma única ação.

A integridade de uma transação depende de 4 propriedades, conhecidas como **ACID**.

Atomicidade-Todas as ações que compõem a unidade de trabalho da transação devem ser concluídas com sucesso, para que seja efetivada. Se durante a transação qualquer ação que constitui unidade de trabalho falhar, a transação inteira deve ser desfeita (rollback). Quando todas as ações são efetuadas com sucesso, a transação pode ser efetivada e persistida em banco (commit).

Consistência-Todas as regras e restrições definidas no banco de dados devem ser obedecidas. Relacionamentos por chaves estrangeiras, checagem de valores para campos restritos ou únicos devem ser obedecidos para que uma transação possa ser completada com sucesso.

CONCEITOS BÁSICOS

DE BANCO DE DADOS, DATAMINING E DATAWAREHOUSE, BANCO DE DADOS RELACIONAL

Isolamento-Cada transação funciona completamente à parte de outras estações. Todas as operações são parte de uma transação única. O princípio é que nenhuma outra transação, operando no mesmo sistema, possa interferir no funcionamento da transação corrente(é um mecanismo de controle). Outras transações não podem visualizar os resultados parciais das operações de uma transação em andamento (ainda em respeito à propriedade da atomicidade).

Durabilidade-Significa que os resultados de uma transação são permanentes e podem ser desfeitos somente por uma transação subsequente. Por exemplo: todos os dados e status relativos a uma transação devem ser armazenados num repositório permanente, não sendo passíveis de falha por uma falha de hardware.

SEGURANÇA EM BANCO DE DADOS

Os bancos de dados são utilizados para armazenar diversos tipos de informações, desde dados sobre uma conta de e-mail até dados importantes da Receita Federal. A segurança do banco de dados herda as mesmas dificuldades que a segurança da informação enfrenta, que é garantir a integridade, a disponibilidade e a confidencialidade. Um Sistema gerenciador de banco de dados deve fornecer mecanismos que auxiliem nesta tarefa.

CONCEITOS BÁSICOS EM BANCO DE DADOS.

TABELAS

Nos modelos de bases de dados relacionais, a **tabela** é um conjunto de dados dispostos em número finito de colunas e número ilimitado de linhas (ou tuplas).

As colunas são tipicamente consideradas os *campos* da tabela, e caracterizam os tipos de dados que deverão constar na tabela (numéricos, alfanuméricos, datas, coordenadas, etc.). O número de linhas pode ser interpretado como o número de combinações de valores dos campos da tabela, e pode conter linhas idênticas, dependendo do objetivo. A forma de referenciar inequivocamente uma única linha é através da utilização de uma chave primária.

Para além do tipo de dados inerente a todas as colunas de uma tabela, algumas podem ter associadas restrições: a unicidade (SQL: UNIQUE), proibição de valores NULL (SQL: NOT NULL), delimitação de valores, etc. Estas restrições impedem que sejam inseridos valores não desejados que comprometam a validade e integridade dos dados. O número de tuplas de uma tabela é virtualmente ilimitado, o que torna as pesquisas por valor potencialmente muito lentas. Para permitir agilizar estas consultas, podem ser associados índices à tabela, que são estruturas de dados independentes da forma e ordem como estão armazenados os dados, embora tenham relação direta com os mesmos. Como consequência, a cada alteração de dados, irá corresponder uma (ou mais) alterações em cada um dos índices, aumentando o esforço necessário ao SGBD para gerir essa alteração, motivo pelo qual os índices não existam naturalmente para cada coluna.

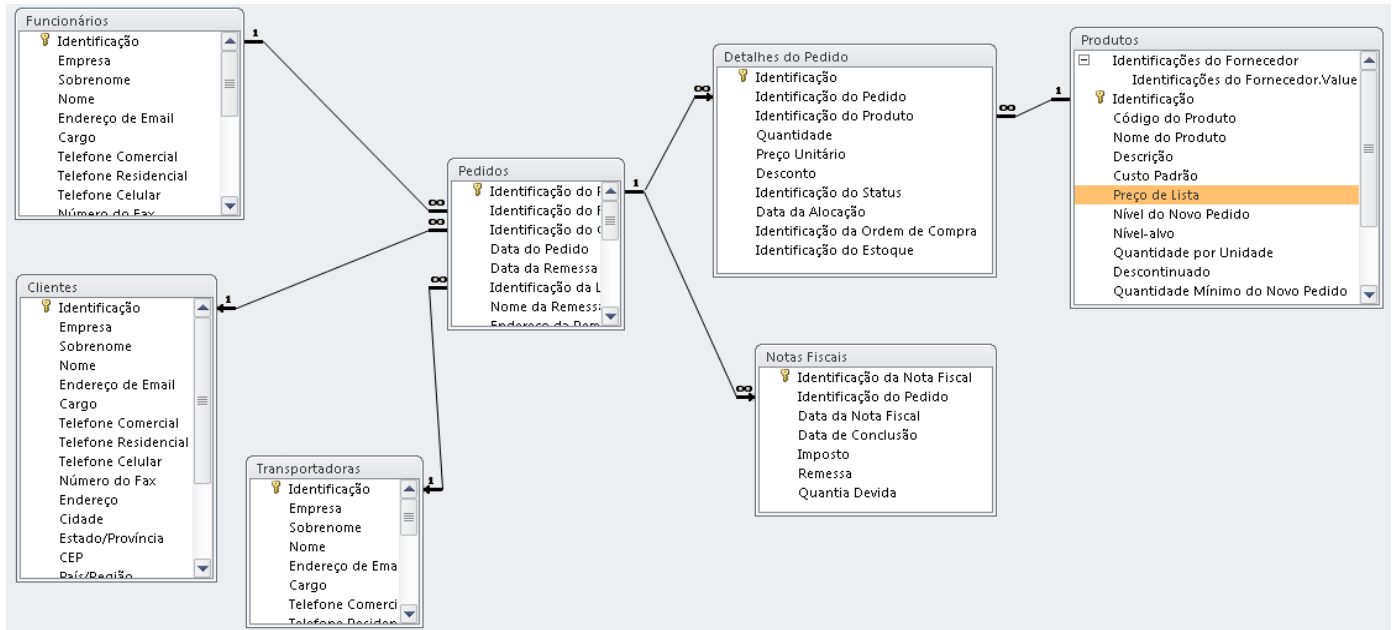
Outro tipo de tabela especial — por não fazer armazenamento de dados — é a vista ou consultas, cujas linhas são determinadas dinamicamente através de uma *query* (consulta) de tabelas reais (que armazenam os dados).

ARQUITETURA DE DADOS

É a estrutura dos componentes de dados de uma organização - considerados sob diferentes níveis de abstração, suas inter-relações, bem como os princípios, diretrizes, normas e padrões que regem seu projeto e evolução ao longo do tempo.

Envolve, portanto, o processo de gerenciamento dos ativos informacionais e o projeto de dados usado para definir uma determinada situação futura, incluindo o subsequente planejamento necessário para alcançar tal estado. É considerada um dos domínios que constituem os pilares da Arquitetura Empresarial. Em um sentido restrito, pode significar também o conjunto das definições de estruturas de dados, relacionamentos e regras comportamentais aplicadas a uma particular solução de Tecnologia da Informação.

CONCEITOS BÁSICOS DE BANCO DE DADOS, DATAMINING E DATAWAREHOUSE, BANCO DE DADOS RELACIONAL



Exemplo de um Banco de Dados Relacional com a representação de suas tabelas e seus relacionamentos.

Cada retângulo seria uma tabela, cada linha interna seriam os campos dessa tabela, as linhas que ligam as tabelas os seus relacionamentos. E os campos que possuem uma imagem de chave no lado esquerdo seriam os campos chaves.

DATA WAREHOUSE

Origem: Wikipédia, a enciclopédia livre.

Também conhecido com **armazém de dados** ou ainda **depósito de dados**, é um sistema de computação utilizado para armazenar informações relativas às atividades de uma organização em bancos de dados, de forma consolidada. O desenho da base de dados favorece os relatórios, a análise de grandes volumes de dados e a obtenção de informações estratégicas que podem facilitar a **tomada de decisão**.

O *data warehouse* possibilita a análise de grandes volumes de dados, coletados dos sistemas transacionais (que fazem as transações da empresa). São as chamadas séries históricas que possibilitam uma melhor análise de eventos passados, oferecendo suporte às tomadas de decisões presentes e a previsão de eventos futuros. Por definição, os dados em um *data warehouse* não são voláteis, ou seja, eles não mudam, salvo quando é necessário fazer correções de dados previamente carregados. Os dados estão disponíveis somente para leitura e não podem ser alterados.

A ferramenta mais popular para exploração de um *data warehouse* é a *Online Analytical Processing OLAP* ou Processo Analítico em Tempo Real, mas muitas outras podem ser usadas.

Os *data warehouse* surgiram como conceito acadêmico na década de 80. Com o amadurecimento dos sistemas de informação empresariais, as necessidades de análise dos dados cresceram paralelamente. Os sistemas OLTP não conseguiam cumprir a tarefa de análise com a simples geração de relatórios. Nesse contexto, a implementação do *data warehouse* passou a se tornar realidade nas grandes corporações. O mercado de ferramentas de *data warehouse*, que faz parte do mercado de *Business Intelligence*, cresceu então, e ferramentas melhores e mais sofisticadas foram desenvolvidas para apoiar a estrutura do *data warehouse* e sua utilização.

Atualmente, por sua capacidade de sumarizar e analisar grandes volumes de dados, o *data warehouse* é o núcleo dos sistemas de informações gerenciais e apoio à decisão das principais soluções de *business intelligence* do mercado.

DATA MINING

CONCEITOS BÁSICOS DE BANCO DE DADOS, DATA MINING E DATAWAREHOUSE, BANCO DE DADOS RELACIONAL



Prospecção de dados ou **mineração de dados** é o processo de explorar grandes quantidades de dados à procura de padrões consistentes, como regras de associação ou sequências temporais, para detectar relacionamentos sistemáticos entre variáveis, detectando assim novos subconjuntos de dados.

Esse é um tópico recente em ciência da computação, mas utiliza várias técnicas da estatística, recuperação de informação, inteligência artificial e reconhecimento de padrões.

A mineração de dados é formada por um conjunto de ferramentas e técnicas que através do uso de algoritmos de aprendizagem ou classificação baseados em redes neurais e estatística, são capazes de explorar um conjunto de dados, extraindo ou ajudando a evidenciar padrões nestes dados e auxiliando na descoberta de conhecimento. Esse conhecimento pode ser apresentado por essas ferramentas de diversas formas: agrupamentos, hipóteses, regras, árvores de decisão, grafos.

O ser humano sempre aprendeu observando padrões, formulando hipóteses e testando-as para descobrir regras. A novidade da era do computador é o volume enorme de dados que não pode mais ser examinado à procura de padrões em um prazo razoável. A solução é instrumentalizar o próprio computador para detectar relações que sejam novas e úteis. A mineração de dados (MD) surge para essa finalidade e pode ser aplicada tanto para a pesquisa científica como para impulsionar a lucratividade da empresa madura, inovadora e competitiva.

Diariamente as empresas acumulam grande volume de dados em seus aplicativos operacionais. São dados brutos que dizem quem comprou o quê, onde, quando e em que quantidade. É a informação vital para o dia-a-dia da empresa. Se fizermos estatística ao final do dia para repor estoques e detectar tendências de compra, estaremos praticando business intelligence (BI). Se analisarmos os dados com estatística de modo mais refinado, à procura de padrões de vinculações entre as variáveis registradas, então estaremos fazendo mineração de dados. Buscamos com a MD conhecer melhor os clientes, seus padrões de consumo e motivações. A MD resgata em organizações grandes o papel do dono atendendo no balcão e conhecendo sua clientela. Através da MD, esses dados agora podem agregar valor às decisões da empresa, sugerir tendências, desvendar particularidades dela e de seu meio ambiente e permitir ações melhor informadas aos seus gestores.

Pode-se então diferenciar o business intelligence (BI) da mineração de dados (MD) como dois patamares distintos de atuação. O primeiro busca subsidiar a empresa com conhecimento novo e útil acerca do seu meio ambiente e funciona no plano estratégico. O Segundo visa obter a partir dos dados operativos brutos, informação útil para subsidiar a tomada de decisão nos escalões médios e altos da empresa e funciona no plano tático.

Etapas da mineração de dados

Os passos fundamentais de uma mineração bem sucedida a partir de fontes de dados (bancos de dados, relatórios, logs de acesso, transações, etc.) consistem de uma limpeza (consistência, preenchimento de informações, remoção de ruído e redundâncias, etc.). Disto nascem os repositórios organizados (*Data Marts* e *Data Warehouses*).

CONCEITOS BÁSICOS DE BANCO DE DADOS, DATAMINING E DATAWAREHOUSE, BANCO DE DADOS RELACIONAL

É a partir deles que se pode selecionar algumas colunas para atravessarem o processo de mineração. Tipicamente, este processo não é o final da história: de forma interativa e frequentemente usando visualização gráfica, um analista refina e conduz o processo até que os padrões apareçam. Observe que todo esse processo parece indicar uma hierarquia, algo que começa em instâncias elementares (embora volumosas) e terminam em um ponto relativamente concentrado.

Encontrar padrões requer que os dados brutos sejam sistematicamente "simplificados" de forma a desconsiderar aquilo que é específico e privilegiar aquilo que é genérico. Faz-se isso porque não parece haver muito conhecimento a extrair de eventos isolados. Uma loja de sua rede que tenha vendido a um cliente uma quantidade impressionante de um determinado produto em uma única data pode apenas significar que esse cliente em particular procurava grande quantidade desse produto naquele exato momento. Mas isso provavelmente não indica nenhuma tendência de mercado.

Localizando padrões

Padrões são unidades de informação que se repetem. A tarefa de localizar padrões não é privilégio da mineração de dados. O cérebro dos seres humanos utiliza-se de processos similares, pois muito do conhecimento que temos em nossa mente é, de certa forma, um processo que depende da localização de padrões. Para exemplificar esses conceitos, vamos propor um breve exercício de indução de regras abstratas. Nosso objetivo é tentar obter alguma expressão genérica para a seguinte sequência:

Sequência original: **ABCXYABCZKABDKCABCTUABEWLABCWO**

Observe atentamente essa sequência de letras e tente encontrar alguma coisa relevante. Veja algumas possibilidades:

Passo 1: A primeira etapa é perceber que existe uma sequência de letras que se repete bastante. Encontramos as sequências "AB" e "ABC" e observamos que elas ocorrem com frequência superior à das outras sequências.

Passo 2: Após determinarmos as sequências "ABC" e "AB", verificamos que elas segmentam o padrão original em diversas unidades independentes:

"ABCXY"
"ABCZK"
"ABDKC"
"ABCTU"
"ABEWL"
"ABCWO"

Passo 3: Fazem-se agora induções, que geram algumas representações genéricas dessas unidades:

"ABC??" "ABD??" "ABE??" e "AB???",

onde '?' representa qualquer letra

No final desse processo, toda a sequência original foi substituída por regras genéricas indutivas, o que simplificou (reduziu) a informação original a algumas expressões simples. Esta explicação é um dos pontos essenciais da mineração de dados, como se pode fazer para extrair certos padrões de dados brutos. Contudo, mais importante do que simplesmente obter essa redução de informação, esse processo nos permite gerar formas de prever futuras ocorrências de padrões.

Exemplo prático

Vamos observar aqui apenas um pequeno exemplo prático do que podemos utilizar com as expressões abstratas genéricas que obtivemos. Uma dessas expressões nos diz que toda vez que encontramos a sequência "AB", podemos inferir que iremos encontrar mais três caracteres e isto completaria um "padrão". Nesta forma abstrata ainda pode ficar difícil de perceber a relevância deste resultado. Por isso vamos usar uma representação mais próxima da realidade.

CONCEITOS BÁSICOS DE BANCO DE DADOS, DATAMINING E DATAWAREHOUSE, BANCO DE DADOS RELACIONAL

Imagine que a letra 'A' esteja representando um item qualquer de um registro comercial. Por exemplo, a letra 'A' poderia significar "aquisição de pão" em uma transação de supermercado. A letra 'B' poderia, por exemplo, significar "aquisição de leite". A letra 'C' é um indicador de que o leite que foi adquirido é do tipo desnatado. É interessante notar que a obtenção de uma regra com as letras "AB" quer dizer, na prática, que toda vez que alguém comprou pão, também comprou leite. Esses dois atributos estão associados e isto foi revelado pelo processo de descoberta de padrões.

Esta associação já nos fará pensar em colocar "leite" e "pão" mais próximos um do outro no supermercado, pois assim estaríamos facilitando a aquisição conjunta desses dois produtos. Mas a coisa pode ir, além disso, bastando continuar nossa exploração da indução.

Suponha que a letra 'X' signifique "manteiga sem sal", e que a letra 'Z' signifique "manteiga com sal". A letra 'T' poderia significar "margarina". Parece que poderíamos tentar unificar todas essas letras através de um único conceito, uma idéia que resuma uma característica essencial de todos esses itens. Introduzimos a letra 'V', que significaria "manteiga/margarina", ou "coisas que passamos no pão". Fizemos uma indução orientada a atributos, substituímos uma série de valores distintos (mas similares) por um nome só.

Ao fazer isso estamos perdendo um pouco das características dos dados originais. Após essa transformação, já não sabemos mais o que é manteiga e o que é margarina. Essa perda de informação é fundamental na indução e é um dos fatores que permite o aparecimento de padrões mais gerais. A vantagem desse procedimento é de que basta codificar a sequência original substituindo a letra 'V' em todos os lugares devidos. Assim fica essa sequência transformada:

ABCVYABCVKABDKCABCVUABEWLABCVO

Daqui, o sistema de mineração de dados irá extrair, entre outras coisas, a expressão "ABCV", que irá revelar algo muito interessante:

A maioria dos usuários que adquiriram pão e leite desnatado também adquiriram manteiga ou margarina.

De posse desta regra, fica fácil imaginar uma disposição nas prateleiras do supermercado para incentivar ainda mais este hábito. Em linguagem mais lógica, pode-se dizer que pão e leite estão associados (implicam) na aquisição de manteiga, isto é,

Pao, Leite → Manteiga

Exemplos Reais

Vestibular PUC-RJ

Utilizando as técnicas da mineração de dados, um programa de obtenção de conhecimento depois de examinar milhares de alunos forneceu a seguinte regra: se o candidato é do sexo feminino, trabalha e teve aprovação com boas notas no vestibular, então não efetivava a matrícula. Estranho, ninguém havia pensado nisso. Mas uma reflexão justifica a regra oferecida pelo programa: de acordo com os costumes do Rio de Janeiro, uma mulher em idade de vestibular, se trabalha é porque precisa, e neste caso deve ter feito inscrição para ingressar na universidade pública gratuita. Se teve boas notas provavelmente foi aprovada na universidade pública onde efetivará matrícula. Claro que há exceções: pessoas que moram em frente à PUC, pessoas mais velhas, de alto poder aquisitivo e que voltaram a estudar por outras razões que ter uma profissão, etc.. Mas a grande maioria obedece à regra anunciada.

Estado civil x cargos de servidores da SEFAZ-AM

Com o uso de data mining foram verificadas correlações entre o estado civil e salários da Secretaria de Fazenda do Estado do Amazonas. Notava-se que cerca de 80% dos servidores de maior poder aquisitivo deste órgão eram divorciados/desquitados, enquanto que em outras instituições, como por exemplo, na Secretaria de Educação (composta em sua maioria por professores), esta média de divorciados/desquitados era inferior a 30%. Longe de parecer coincidência, os dados sugerem que servidores com maior poder aquisitivo se envolvam com relações extraconjugais, resultando geralmente em desfazimento do casamento.

CONHECIMENTOS BÁSICOS DE AMBIENTE DE SERVIDORES

Estrutura de servidores físicos e virtualizados

Os **servidores** – são máquinas que hospedam arquivos e aplicativos em redes de computadores - precisam ser poderosos. Algumas dessas máquinas dispõem de unidades de processamento central (CPUs) com múltiplos processadores que dão a elas a capacidade de executar com facilidade tarefas complexas. Os administradores de redes de computação geralmente dedicam cada servidor a um aplicativo ou a uma tarefa específica. Muitas dessas tarefas não funcionam bem quando realizadas em companhia de outras - cada uma precisa de uma máquina exclusiva. Uma aplicação por servidor também torna mais fácil encontrar os problemas quando eles surgem. É uma maneira simples de criar uma rede de computadores enxuta, do ponto de vista técnico.

Mas existem problemas quanto a essa abordagem. Um deles é que ela não aproveita o poder de processamento dos servidores modernos. A maioria dos servidores emprega apenas uma fração de sua capacidade de processamento. Outro problema é que, à medida que uma rede de computadores se torna maior e mais complexa, os servidores começam a ocupar muito espaço físico. Um **centro de dados** pode ficar repleto de **rack**s de servidores que consomem muita energia e geram calor.

Os **servidores virtuais** representam uma tentativa de resolver definitivamente essas duas questões. Por meio de um software especializado, o administrador pode converter um servidor físico em múltiplas máquinas virtuais. Cada servidor virtual funciona como um aparelho físico único, capaz de operar seu próprio **sistema operacional** (SO). Em teoria, seria possível criar número suficiente de servidores virtuais para usar todo o poder de processamento de uma máquina, ainda que isso nem sempre represente a melhor ideia em termos práticos.

A virtualização não representa um conceito novo. Há décadas, os cientistas da computação vêm criando máquinas virtuais em supercomputadores, mas há apenas alguns anos a virtualização se tornou possível em servidores. No mundo da tecnologia da informação (TI), a virtualização de servidores é assunto quente. A tecnologia é jovem e diversas empresas oferecem diferentes abordagens.

Virtualização de Servidor

A Virtualização de Servidor é a técnica de execução de um ou mais servidores virtuais sobre um servidor físico. Permite maior densidade de utilização de recursos (hardware, espaço e etc), enquanto permite que isolamento e segurança sejam mantidos. Com a Virtualização de Servidor, conquista-se os seguintes benefícios:

Consolidação de Servidores: Muitos servidores implantados pelas organizações são subutilizados. Implantando múltiplos servidores em um número menor de servidores físicos, é possível aumentar a utilização média de recursos dos servidores, enquanto diminui o número de máquinas. Na maioria das organizações, consolidar os servidores com Virtualização de Servidores diminui os gastos com eletricidade, consumo de espaço e etc.

Isolamento de Aplicação ou Serviço: Com a criação de máquinas virtuais isoladas, a execução dos serviços e aplicações é feita em Sistemas Operacionais diferentes. Isso previne que uma aplicação afete outra quando você faz uma atualização ou mudança. Isso se torna melhor do que executar diversas aplicações em um único Sistema Operacional.

Implantação de Servidores Simplificada: Com a criação de imagens padrão de servidores virtuais, você pode implantar máquinas virtuais de forma muito mais simples. Como você está implementando um servidor virtual, você também não precisa fazer aquisição de um novo Hardware, e localizar espaço e energia elétrica em um Data Center. (Observando sempre a utilização de recursos compartilhados dentro de um Host, você pode ter que adquirir um novo Hardware para executar suas Máquinas Virtuais)

Maior disponibilidade de Aplicações e Serviços: Como a aplicação ou serviço não está mais conectado diretamente a um hardware específico, é mais fácil assegurar disponibilidade e recuperação. Algumas tecnologias permitem, inclusive, migrar uma máquina virtual de um host a outro host sem interrupção da máquina virtual.

Múltiplos Sistemas Operacionais podem ser executados uma única plataforma: Com a virtualização, é possível utilizar diferentes Sistemas Operacionais em um único servidor físico, como Windows Server 2003 e Windows Server 2008 R2 e até mesmo Linux.

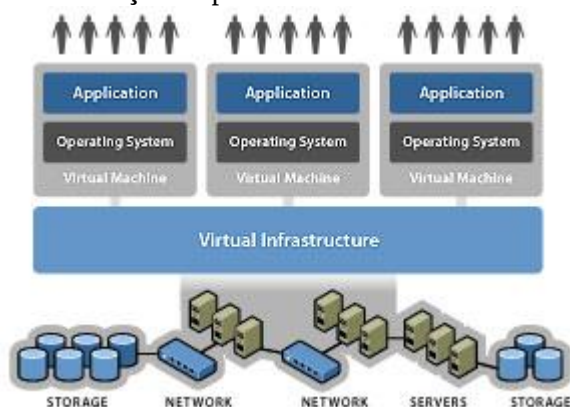
Conceito de Virtualização

Virtualização é a reorganização de ambientes operacionais de servidores físicos em ambientes de servidores virtualizados. Isso é possível através da emulação de hardwares por meio de software, obtendo-se virtual machines.

CONHECIMENTOS BÁSICOS DE AMBIENTE DE SERVIDORES

Estrutura de servidores físicos e virtualizados

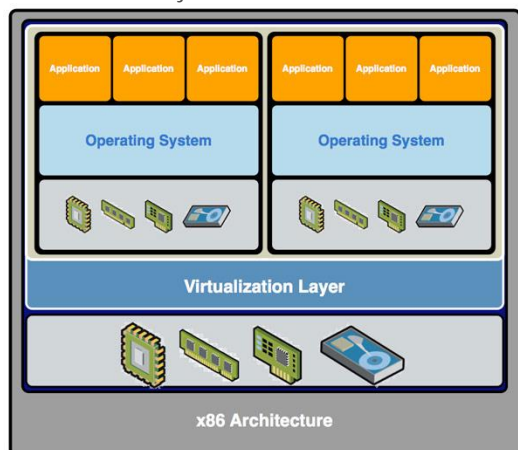
Virtualização na prática



Utilizando-se de camadas adicionais de software (hypervisors) entre as máquinas virtuais e o servidor para compartilhamento de hardware, a **Virtualização** permite que múltiplos sistemas operacionais (virtual machines) possam ser executados em um único servidor físico. Cada máquina virtual possuirá seu sistema operacional totalmente independente e possuirá desempenho muito similar a uma máquina física. Ainda através da **Virtualização** cada uma das virtual machines podem oferecer serviços de rede interconectar-se (virtualmente) através de interfaces de rede, switches, roteadores e firewalls virtuais.

Alocação de Recursos

Ao contrário dos servidores convencionais, ao criar uma máquina virtual, aloca-se recursos como memória, espaço de disco, processamento por meio de dispositivos lógicos, de acordo com a necessidade específica da aplicação à qual determinada virtual machine se destine. A **Virtualização** possibilita a utilização do Monitor de máquina virtual. O Monitor pode ser programado para, em caso de determinada virtual machine necessite recursos além do que a ela foi destinado, realocar automaticamente recursos ociosos do servidor virtualizado em que a virtual machine se encontre ou de outros servidores virtualizados para manter a estabilidade de desempenho e atender as expectativas de níveis de serviço.



Bibliografia

- Data Mining Blog – Blog sobre Data mining e Data ware house <http://www.fp2.com.br/blog/>
- Wikipedia – Enciclopédia Livre <http://pt.wikipedia.org/>
- NCE-UFRJ – Instituto Tércio Pacitti de Aplicações e Pesquisas Computacionais <http://portal.nce.ufrj.br/>
- PUC-RS – Conceitos básicos de programação - Márcio Sarroglia Pinho - <http://www.inf.pucrs.br/>