

INPE-414-LAFE

DPD

UMA LINGUAGEM DE BUSCA
PARA
SISTEMAS DE RECUPERAÇÃO DE INFORMAÇÃO

Julho 1973

cc.: 60



SERVIÇO PÚBLICO FEDERAL
CONSELHO NACIONAL DE PESQUISAS
INSTITUTO DE PESQUISAS ESPACIAIS
São José dos Campos - Estado de S. Paulo - Brasil

*UMA LINGUAGEM DE BUSCA
PARA
SISTEMAS DE RECUPERAÇÃO DE INFORMAÇÃO*

Este trabalho foi apresentado, como parte dos requisitos necessários à obtenção do grau de Mestre em Computação Aplicada, pelos Engenheiros Iberê Lucio Ronchetti Teixeira e Arry Carlos Buss Filho deste Instituto e a presente publicação foi autorizada pelo abaixo assinado.

F de Mendonça
Fernando de Mendonça
Diretor Geral



PRESIDÊNCIA DA REPÚBLICA
CONSELHO NACIONAL DE PESQUISAS
INSTITUTO DE PESQUISAS ESPACIAIS
São José dos Campos - Estado de S. Paulo - Brasil

FORMULÁRIO PG-012

AVALIAÇÃO DE PROJETO COLETIVO

1. Aluno: Arry Carlos Buss Filho
2. Assunto do Trabalho: Recuperação de Informação
3. Período: 2º/1973 Projeto: DPD
4. Orientador: Dr. Nelscon C. Machado Ass. do Avaliador: *[Assinatura]*
5. Setor de Pós-Graduação: Computação Aplicada
6. Membro: Dr. Dutta-Roy

INSTRUÇÕES AO AVALIADOR

1. A apreciação de cada fator deverá ser efetuada pela colocação de uma cruz (+) no losango correspondente da esquerda. Caso o avaliador julgar que a apreciação está situada entre dois fatores subsequentes, deverá assinalar o losango correspondente da direita.
2. O avaliador deverá julgar em relação a cada fator separadamente, sem considerar sua impressão geral do conjunto.
3. É importante basear as avaliações em fatos e não em opiniões.
4. É necessário que o orientador tome sua própria decisão, não se deixando influenciar pela decisão dos outros.
5. Deve-se procurar evitar cair nos erros geralmente cometidos pelo avaliador:
 - tendência natural para elevar a categoria por benevolência;
 - tendência a que a impressão do conjunto se sobreponha a cada qualidade particular;
 - tendência de deixar-se levar pelo desejo de favorecer consciente ou inconscientemente, o colaborador.
6. A avaliação deve ser feita imediatamente após a apresentação de cada grupo, perante a Banca Examinadora.
7. A avaliação deve ser feita individualmente, por cada membro da Banca Examinadora, e para cada elemento integrante do grupo.
8. Após a avaliação os membros da Banca Examinadora, devem remeter as suas Fôlhas de Avaliação diretamente para a Divisão de Ensino, para o cálculo da média e aferição de créditos.



PRESIDÊNCIA DA REPÚBLICA
CONSELHO NACIONAL DE PESQUISAS
INSTITUTO DE PESQUISAS ESPACIAIS
São José dos Campos - Estado de S. Paulo - Brasil

FORMULÁRIO PG-012

AVALIAÇÃO DE PROJETO COLETIVO

1. Aluno: Arry Carlos Buss F9
2. Assunto do Trabalho: Recuperação de Informação
3. Período: 2º / 19 73 Projeto: DPD
4. Orientador: Dr. Nelson C. Machado Ass. do Avaliador: Ricardo Palmeira
5. Setor de Pós-Graduação: Computação Aplicada
6. Membro: Dr. Ricardo A.R. Palmeira

INSTRUÇÕES AO AVALIADOR

1. A apreciação de cada fator deverá ser efetuada pela colocação de uma cruz (+) no losango correspondente da esquerda. Caso o avaliador julgar que a apreciação está situada entre dois fatores subsequentes, deverá assinalar o losango correspondente da direita.
2. O avaliador deverá julgar em relação a cada fator separadamente, sem considerar sua impressão geral do conjunto.
3. É importante basear as avaliações em fatos e não em opiniões.
4. É necessário que o orientador tome sua própria decisão, não se deixando influenciar pela decisão dos outros.
5. Deve-se procurar evitar cair nos erros geralmente cometidos pelo avaliador:
 - tendência natural para elevar a categoria por benevolência;
 - tendência a que a impressão do conjunto se sobreponha a cada qualidade particular;
 - tendência de deixar-se levar pelo desejo de favorecer consciente ou inconscientemente, o colaborador.
6. A avaliação deve ser feita imediatamente após a apresentação de cada grupo, perante a Banca Examinadora.
7. A avaliação deve ser feita individualmente, por cada membro da Banca Examinadora, e para cada elemento integrante do grupo.
8. Após a avaliação os membros da Banca Examinadora, devem remeter as suas Fôlhas de Avaliação diretamente para a Divisão de Ensino, para o cálculo da média e aferição de créditos.



PRESIDÊNCIA DA REPÚBLICA
CONSELHO NACIONAL DE PESQUISAS
INSTITUTO DE PESQUISAS ESPACIAIS
São José dos Campos - Estado de S. Paulo - Brasil

FORMULÁRIO PG-012

AVALIAÇÃO DE PROJETO COLETIVO

1. Aluno: Arry Carlos Buss Fº
2. Assunto do Trabalho: Recuperação de Informação
3. Período: 2º/19 73 Projeto: DPD
4. Orientador: Dr. Nelson C. Machado Ass. do Avaliador: *Jaf*
5. Setor de Pós-Graduação: Computação Aplicada
6. Membro: Dr. Norberto W. Dachs

INSTRUÇÕES AO AVALIADOR

1. A apreciação de cada fator deverá ser efetuada pela colocação de uma cruz (+) no losango correspondente da esquerda. Caso o avaliador julgar que a apreciação está situada entre dois fatores subsequentes, deverá assinalar o losango correspondente da direita.
2. O avaliador deverá julgar em relação a cada fator separadamente, sem considerar sua impressão geral do conjunto.
3. É importante basear as avaliações em fatos e não em opiniões.
4. É necessário que o orientador tome sua própria decisão, não se deixando influenciar pela decisão dos outros.
5. Deve-se procurar evitar cair nos erros geralmente cometidos pelo avaliador:
 - tendência natural para elevar a categoria por benevolência;
 - tendência a que a impressão do conjunto se sobreponha a cada qualidade particular;
 - tendência de deixar-se levar pelo desejo de favorecer consciente ou inconscientemente, o colaborador.
6. A avaliação deve ser feita imediatamente após a apresentação de cada grupo, perante a Banca Examinadora.
7. A avaliação deve ser feita individualmente, por cada membro da Banca Examinadora, e para cada elemento integrante do grupo.
8. Após a avaliação os membros da Banca Examinadora, devem remeter as suas Fôlhas de Avaliação diretamente para a Divisão de Ensino, para o cálculo da média e aferição de créditos.



PRESIDÊNCIA DA REPÚBLICA
CONSELHO NACIONAL DE PESQUISAS
INSTITUTO DE PESQUISAS ESPACIAIS
São José dos Campos - Estado de S. Paulo - Brasil

FORMULÁRIO PG-012

AVALIAÇÃO DE PROJETO COLETIVO

1. Aluno: Arry Carlos Buss Fº
2. Assunto do Trabalho: Recuperação de Informação
3. Período: 2º / 19 73 Projeto: DPD
4. Orientador: Dr. Nelson C. Machado Ass. do Avaliador: N. C. Machado
5. Setor de Pós-Graduação: Computação Aplicada
6. Membro: Dr. Nelson C. Machado

INSTRUÇÕES AO AVALIADOR

1. A apreciação de cada fator deverá ser efetuada pela colocação de uma cruz (+) no losango correspondente da esquerda. Caso o avaliador julgar que a apreciação está situada entre dois fatores subsequentes, deverá assinalar o losango correspondente da direita.
2. O avaliador deverá julgar em relação a cada fator separadamente, sem considerar sua impressão geral do conjunto.
3. É importante basear as avaliações em fatos e não em opiniões.
4. É necessário que o orientador tome sua própria decisão, não se deixando influenciar pela decisão dos outros.
5. Deve-se procurar evitar cair nos erros geralmente cometidos pelo avaliador:
 - tendência natural para elevar a categoria por benevolência;
 - tendência a que a impressão do conjunto se sobreponha a cada qualidade particular;
 - tendência de deixar-se levar pelo desejo de favorecer consciente ou inconscientemente, o colaborador.
6. A avaliação deve ser feita imediatamente após a apresentação de cada grupo, perante a Banca Examinadora.
7. A avaliação deve ser feita individualmente, por cada membro da Banca Examinadora, e para cada elemento integrante do grupo.
8. Após a avaliação os membros da Banca Examinadora, devem remeter as suas Fôlhas de Avaliação diretamente para a Divisão de Ensino, para o cálculo da média e aferição de créditos.



PRESIDÊNCIA DA REPÚBLICA
CONSELHO NACIONAL DE PESQUISAS
INSTITUTO DE PESQUISAS ESPACIAIS
São José dos Campos - Estado de S. Paulo - Brasil

FORMULÁRIO PG-012

AVALIAÇÃO DE PROJETO COLETIVO

1. Aluno: Arry Carlos Buss F9
2. Assunto do Trabalho: Recuperação de Informação
3. Período: 299 73 Projeto: DPD
4. Orientador: Dr. Nelson C. Machado Ass. do Avaliador: Amir Felipe
5. Setor de Pós-Graduação: Computação Aplicada
6. Membro: Eng. Almir Paz de Lima

INSTRUÇÕES AO AVALIADOR

1. A apreciação de cada fator deverá ser efetuada pela colocação de uma cruz (+) no losango correspondente da esquerda. Caso o avaliador julgar que a apreciação está situada entre dois fatores subsequentes, deverá assinalar o losango correspondente da direita.
2. O avaliador deverá julgar em relação a cada fator separadamente, sem considerar sua impressão geral do conjunto.
3. É importante basear as avaliações em fatos e não em opiniões.
4. É necessário que o orientador tome sua própria decisão, não se deixando influenciar pela decisão dos outros.
5. Deve-se procurar evitar cair nos erros geralmente cometidos pelo avaliador:
 - tendência natural para elevar a categoria por benevolência;
 - tendência a que a impressão do conjunto se sobreponha a cada qualidade particular;
 - tendência de deixar-se levar pelo desejo de favorecer consciente ou inconscientemente, o colaborador.
6. A avaliação deve ser feita imediatamente após a apresentação de cada grupo, perante a Banca Examinadora.
7. A avaliação deve ser feita individualmente, por cada membro da Banca Examinadora, e para cada elemento integrante do grupo.
8. Após a avaliação os membros da Banca Examinadora, devem remeter as suas Fôlhas de Avaliação diretamente para a Divisão de Ensino, para o cálculo da média e aferição de créditos.



PRESIDÊNCIA DA REPÚBLICA
CONSELHO NACIONAL DE PESQUISAS
INSTITUTO DE PESQUISAS ESPACIAIS
São José dos Campos - Estado de S. Paulo - Brasil

FORMULÁRIO PG-012

AValiação DE PROJETO COLETIVO

1. Aluno: Arry Carlos Buss Filho
2. Assunto do Trabalho: Recuperação de Informação
3. Período: 29/19 73 Projeto: DPD
4. Orientador: Dr. Nelscon C. Machado Ass. do Avaliador:
5. Setor de Pós-Graduação: Computação Aplicada
6. Membro: Dr. Luiz Gylvan Meira Fº

INSTRUÇÕES AO AVALIADOR

1. A apreciação de cada fator deverá ser efetuada pela colocação de uma cruz (+) no losango correspondente da esquerda. Caso o avaliador julgar que a apreciação está situada entre dois fatores subsequentes, deverá assinalar o losango correspondente da direita.
2. O avaliador deverá julgar em relação a cada fator separadamente, sem considerar sua impressão geral do conjunto.
3. É importante basear as avaliações em fatos e não em opiniões.
4. É necessário que o orientador tome sua própria decisão, não se deixando influenciar pela decisão dos outros.
5. Deve-se procurar evitar cair nos erros geralmente cometidos pelo avaliador:
 - tendência natural para elevar a categoria por benevolência;
 - tendência a que a impressão do conjunto se sobreponha a cada qualidade particular;
 - tendência de deixar-se levar pelo desejo de favorecer consciente ou inconscientemente, o colaborador.
6. A avaliação deve ser feita imediatamente após a apresentação de cada grupo, perante a Banca Examinadora.
7. A avaliação deve ser feita individualmente, por cada membro da Banca Examinadora, e para cada elemento integrante do grupo.
8. Após a avaliação os membros da Banca Examinadora, devem remeter as suas Fôlhas de Avaliação diretamente para a Divisão de Ensino, para o cálculo da média e aferição de créditos.

AGRADECIMENTOS

Dentre as inúmeras pessoas, que, de uma maneira ou outra, contribuíram para a realização desse trabalho, não podemos deixar de mencionar as seguintes, às quais somos particularmente gratos:

Nosso orientador, Dr. Nelson Castro Machado, dedicou muitas horas de sua já congestionada agenda à cansativas leituras dos originais. Suas revisões, bem como inúmeras outras sugestões e conselhos tiveram a propriedade de transformar este trabalho, dando-lhe a necessária coerência e fazendo-o cientificamente apresentável.

Ao Instituto Brasileiro de Bibliografia e Documentação, através de sua presidente, D. Hagar Espanha Gomes, agradecemos a permissão para a utilização dos dados do Siabe, matéria prima que muito contribuiu para o desenvolvimento do SIRIUS.

Ao Dr. A. Dutta-Roy e a Sra. Yone Chastinet, pela leitura e crítica dos originais.

A qualidade da apresentação do trabalho é devida à dedicação da Srta. Maria Lúcia dos Santos, que, incansavelmente, datilografou os manuscritos e suas correções, dando-lhes forma final.

RESUMO

Os atuais sistemas de recuperação de informação ressentem-se, em geral, de um método simples de consulta, que seja direta e facilmente utilizado pelo usuário. O método de recuperação aqui apresentado baseia-se numa linguagem de busca, através da qual o usuário codifica sua pesquisa, manipulando palavras-chave e/ou descritores que indexam o material a ser recuperado. Desta forma, simplifica-se o enunciado dos termos de interesse, bem como permite-se ao usuário associá-los entre si por meio de operações lógicas, do tipo OU, E e MAS NÃO.

Embora orientado primordialmente para a recuperação de referências bibliográficas, o sistema criado pode ser aplicado a qualquer conjunto de dados que admita a indexação por palavra-chave ou descritor. Além disso, permite que a indexação seja feita sobre várias e diferentes características dos dados a recuperar; por exemplo, para as referências bibliográficas é possível a recuperação por nome do autor e por palavras do título, além de descritores.

ABSTRACT

The current information retrieval systems usually miss a simple method of consultation, directly and easily employed by the user. The retrieval method presented here is based in a query language, through which the user codes his search, dealing with key-words and descriptors. In this manner, the listing of the terms of interest is quite simplified, and the user has the ability to associated them by means of logical operations, like OR, AND and AND NOT.

Although primarily oriented towards bibliographic references this system may be applied to any data that can be indexed by key-words or descriptors. This indexation may be done over different types of characteristics of the data; for example, it is possible to retrieve any bibliographic reference by author's name, key-words from the title and by descriptors.

INDICE

CAPÍTULO I

INTRODUÇÃO	1
------------------	---

CAPÍTULO II

CONSIDERAÇÕES GERAIS

2.1 - Conceituação do problema	5
2.2 - O conceito de palavra-chave	6
2.3 - Necessidade de uma linguagem de busca	9
2.4 - Aspectos da Implementação	9

CAPÍTULO III

A LINGUAGEM DE BUSCA

3.1 - Introdução	13
3.2 - Conjunto de caracteres disponíveis	13
3.3 - Componentes básicos	14
3.4 - Delimitadores	15
3.5 - Expressões de busca	15
3.6 - Operações de busca	16
3.7 - Símbolos alternativos para os operadores	19
3.8 - Precedencia dos operadores	19
3.9 - Operandos de busca	21
3.9.1 - Nomes de autores	21
3.9.2 - Termos auxiliares	23
3.10- Prefixos comuns	24

CAPÍTULO IV

A CODIFICAÇÃO DOS PROGRAMAS DE BUSCA

4.1 - Programas de busca	25
4.2 - Campos do cartão-fonte	25
4.3 - Descrição detalhada dos diversos comandos	26
4.3.1 - Comentários	26
4.3.2 - Cartão de identificação	26
4.3.3 - Definição de termos auxiliares	27
4.3.4 - Expressão desejada	27
4.4 - Outras considerações sobre as expressões de busca ...	28
4.5 - Estrutura de um programa	29

CAPÍTULO V

ALGORITMOS DE RECUPERAÇÃO

5.1 - Introdução	33
5.2 - A referência bibliográfica típica	34
5.3 - As referencias bibliográficas em disco	35
5.4 - Arquivo de palavras-chave	37
5.5 - Pesquisa binária	39
5.6 - Algoritmo para pesquisa binária	40
5.7 - Algoritmos para as operações OU, E e MAS NAO	42
5.7.1 - Algoritmo para a operação OU	43
5.7.2 - Algoritmo para a operação E	44
5.7.3 - Algoritmo para a operação MAS NAO	44

CAPÍTULO VI

INTERPRETAÇÃO E EXECUÇÃO DOS COMANDOS DE BUSCA

6.1 - Introdução	47
------------------------	----

6.2 - Análise sintática	47
6.3 - Aspecto interno de um operando	50
6.4 - A tabela de termos	51
6.5 - Os reconhecedores	52
6.5.1 - Reconhecedor de termo auxiliar	52
6.5.2 - Reconhecedor de termo chave	53
6.5.3 - Reconhecedor de operadores	54
6.6 - Avaliação da expressão de busca	54
6.7 - A materialização das operações	58
6.8 - Obtenção das listas operando e criação da lista resultado.....	60
6.9 - Tempos de execução	61
6.10- Impressão das referências recuperadas	63

CAPÍTULO VII

CONCLUSÕES	65
------------------	----

<u>BIBLIOGRAFIA</u>	69
---------------------------	----

APÊNDICES

A - Representação dos nomes de autor	71
B - Exemplos completos de pesquisas	73
C - Erros detetados e as mensagens correspondentes	97
D - Sintaxe da linguagem de busca em notação Backus-Naur	103
E - Propriedades da operação MAS NAO	109

CAPÍTULO I

INTRODUÇÃO

Os dois parágrafos abaixo pertencem a um trabalho apresentado em 1969 ao II Congresso Regional de Documentação (referência 18), e dão uma idéia da enorme quantidade de informação que diariamente vem à luz.

Conforme dados estatísticos, o conhecimento duplicou, pela primeira vez, no ano de 1750; pela segunda vez, no ano de 1900 (150 anos depois); pela terceira vez, em 1950 (50 anos depois). Daí para cá, levou apenas 10 anos para se duplicar (em 1960). Atualmente, peritos no assunto calculam uma duplicação anual.

A produção bibliográfica, por sua vez, também é vertiginosamente crescente: cerca de 1.000 títulos novos de livros são publicados diariamente. Se acrescentarmos a estes: 33.000 jornais, 70.000 revistas e mais outros tipos de material especial, tais como, mapas, músicas, filmes, diafilmes, discos etc., o total montará a 10.000.000 de títulos identificáveis separadamente.

Esta "explosão de conhecimentos e informações", de per se desejável, apresenta uma série de problemas: como evitar redundância, não reescrevendo ou reinventando coisas já escritas ou já inventadas? como ficar a par do que é publicado, dentro de uma certa área ou ramo do conhecimento? como conhecer o que escreve uma ou mais pessoas?

Pesquisadores voltados para a solução destes problemas têm procurado utilizar computadores eletrônicos - outro fenômeno da segunda metade do século XX - desde o seu aparecimento. Dos mais antigos foi o sistema implantado na Central Intelligence Agency-CIA- para a confecção de

Índices permutados. Data de 1953 (referência 1).

Em 1958, Hans Peter Luhn cunhou o termo KWIC (key-word in context), inventando um índice permutado de palavras do título. O trabalho de Luhn, considerado como fundamental, foi publicado em 1959 (referência 2), e dele derivaram muitos outros. Marguerite Fischer expõe uma extensa lista destas derivações da idéia de Luhn, tendo todas como ponto em comum o uso do computador digital (referência 1).

Em 1966, a Biblioteca do Congresso dos Estados Unidos deu início ao projeto MARC (Machine Readable Cataloging), de catalogação através de computadores. O sucesso e a experiência ganha levaram à modificação do formato dos registros em fita magnética, surgindo o MARC II (referências 3 e 4).

Outras áreas relacionadas, nas quais os computadores desempenham papel preponderante, são a disseminação seletiva de informação e o índice de citações. A primeira, também começada por Luhn, em 1960, a partir de seu índice KWIC, e da qual um dos melhores exemplos é o sistema montado pela National Science Library, de Ottawa, Canadá (referência 5), faz chegar ao usuário todas as novas publicações que cobrem assuntos previamente estipulados por este, em bases automáticas e periódicas. O índice de citações, introduzido por E. Garfield, permite uma pesquisa bibliográfica para o futuro, isto é, a partir de uma referência básica é possível encontrar publicações posteriores, que a citam, permitindo seguir a evolução de uma idéia (referência 6).

Finalmente, a aplicação mais espetacular talvez seja a dos centros de documentação computadorizados, também conhecidos como bancos de dados. Tais centros começaram a surgir a partir de 1958, quando a General Electric, com um computador IBM 704, criou um sistema de documentação no campo da engenharia aeronáutica (referência 7).

Os bancos de dados se popularizaram com extraordinária ra

pidez. Entre os mais famosos podem ser citados o sistema da National Library of Medicine, o MEDLARS, sigla de Medical Literature Analysis and Retrieval System, e o sistema construído pela Siemens para os Jogos Olímpicos de Munique. O interesse pelos bancos de dados é tão grande que justifica o desenvolvimento e a comercialização de sistemas desse tipo, pelos fabricantes de equipamento de processamento de dados. Exemplos de tais sistemas são o Aquarius (IBM) e o Lidia (Siemens). Uma aplicação do sistema Aquarius particularmente bem conhecida no Brasil é o sistema de recuperação de informações do Senado Federal (referência 11).

Como se depreende do exposto acima, a utilização de computadores para a automação de processos bibliográficos e de documentação é idéia antiga. Dois pontos de vista distintos se observam nitidamente nos trabalhos relativos a sistemas desse tipo. O primeiro deles é o dos documentalistas. A estes interessa especialmente o critério de indexação, a precisar melhor o conteúdo da informação. Não se dá ênfase aos trabalhos de implementação. O segundo ponto de vista é o do cientista de computação, o qual está voltado para a construção de novos e mais eficientes métodos de armazenagem e de busca, o desenvolvimento de dispositivos mais rápidos ou mais eficientes. A utilização de suas descobertas em algum sistema prático, real, não é fundamental.

Entretanto, é opinião dos autores que é oportuna a introdução de um terceiro ponto de vista no estudo de sistemas de recuperação de informação, qual seja, o do usuário comum do sistema, não particularmente versado em biblioteconomia ou em ciência da computação, mas que deseja simplesmente utilizar o sistema para efetuar uma pesquisa bibliográfica de maneira rápida e simples. A este usuário não interessam os detalhes de catalogação ou indexação nem a elegância dos algoritmos de busca. Interessa-lhe, isso sim, uma linguagem eficiente que lhe permita comunicar-se com o sistema de maneira natural e precisa. Quanto a isso, a maioria dos sistemas existentes é relativamente omissa. Ao usuário são fornecidos apenas os meios para fazer as consultas mais simples tirando-lhe a visão de conjunto de sua própria pesquisa.

A solução que imediatamente se apresenta é a criação de uma linguagem de alto nível, tal como Fortran, Algol ou PL/I, que tanto fizeram por popularizar o uso dos computadores por pessoas não especializadas. Parece claro que a solução ideal não será proposta nem pelo cientista de computação nem pelo documentalista, mas sim por uma combinação de ambos, já que se procura fundir os conhecimentos de máquina de um com os conhecimentos de documentação de outro. A conjugação de esforços desses dois especialistas permitirá a introdução de linguagens de busca cada vez mais poderosas e sofisticadas, as quais permitirão que o usuário prescindir de intermediários para a consecução de seus objetivos.

Damos, com o presente trabalho, um passo nessa direção. Neste apresentamos uma linguagem de busca, chamada SIRIUS, ainda bastante simples, como simples é também a estrutura dos dados que pretende recuperar. Apesar disto, e de se tratar de uma primeira implementação, o sistema já é suficientemente complexo o que, paradoxalmente, facilita consideravelmente a sua utilização. Aliam-se, assim, técnicas já bastante conhecidas em teoria da compilação e técnicas usuais em recuperação.

CAPÍTULO II

CONSIDERAÇÕES GERAIS

2.1 - CONCEITUAÇÃO DO PROBLEMA

O presente trabalho apresenta um método de recuperação de informação, na forma de programas de computador, capaz de recuperar, de um conjunto de informações que admitam a aplicação do conceito de palavra-chave (e/ou descritores), aqueles elementos que se enquadrem dentro de uma determinada categoria, fornecida pelo interessado na recuperação.

Ao longo deste trabalho, consideramos que as informações que queremos recuperar são referências bibliográficas, isto é, artigos publicados em periódicos ou apresentados em congressos, monografias, relatórios de pesquisas, etc.

Dentre os motivos que nos levaram a pensar primordialmente em referências bibliográficas como material a ser recuperado, três deles merecem citação especial.

Inicialmente, diríamos que o processo de palavras chave é adotado universalmente, partindo-se da idéia de que elas podem representar o conteúdo de um documento, possibilitando sua recuperação. Atualmente, os periódicos de maior importância em cada área, exigem de seus colaboradores a atribuição de palavras chave aos títulos de seus trabalhos, prevendo uma futura recuperação por estes mesmos termos, o que pode ser exemplificado através dos índices KWIC, de largo emprego.

Além disso, a produção intelectual em ciência e tecnologia tem crescido em tal proporção, que a recuperação das informações de que

tratam, transformou-se em um problema de solução extremamente complexa. Fazemos, com este trabalho, uma contribuição para melhorar a situação, e justamente através da ligação entre palavras chave e referências bibliográficas.

Por último, "last but not least", um motivo de ordem prática. Nos últimos quatro anos, tem o INPE, através de sua Divisão de Processamento de Dados, auxiliado o IBBD - Instituto Brasileiro de Bibliografia e Documentação - na preparação de suas bibliografias nacionais especializadas, elaborando, eletronicamente, os Índices KWIC e de autores que aparecem naquelas bibliografias. Recentemente, o IBBD unificou suas bibliografias lançando o SIABE - Sistema Integrado de Automação de Bibliografias Especializadas (referência 10). Reune o SIABE, presentemente, mais de 50.000 referências bibliográficas, com largo emprego do processo de palavras chave no contexto, e também, quando necessário, enriquecendo seus títulos através de descritores, numa tentativa de maior profundidade de indexação. Essa massa de informação constitui-se não somente em valioso acervo, como sua organização muito simplificou a preparação dos dados para os programas do nosso sistema de recuperação.

Não se deve pensar, porém, que apenas referências bibliográficas podem ser recuperadas por palavras chave, através dos programas elaborados. Outras situações são passíveis de solução através do mesmo tipo de abordagem. Por exemplo, o responsável por uma seção de pessoal pode querer saber quais são, dentre seus funcionários, aqueles que sejam casados, engenheiros, brasileiros e com menos de quarenta anos de idade. Claramente, este é um problema de recuperação de informação a partir de palavras chave. Resolvido o problema da adaptação do cadastro de funcionários ao formato correto para a recuperação, os mesmos programas indicarão os funcionários que preenchem as características desejadas.

2.2 - O CONCEITO DE PALAVRA CHAVE.

Por palavra chave entende-se uma palavra ou frase relativamente curta, que descreve, de alguma forma, características daquilo a que se refere. Em se tratando de trabalhos publicados, as palavras chave geralmente dizem respeito ao assunto ou assuntos discutidos, direta ou indiretamente. Basicamente, pretende-se dar, através das palavras chave, idéia razoavelmente precisa sobre o conteúdo do trabalho.

Para nosso trabalho, por exemplo, caberiam as seguintes palavras chave, algumas das quais constam do título e outras poderiam ser usadas como descritores, para bem caracterizar o assunto do seu texto: recuperação, informação, recuperação de informação, computadores, linguagem de programação, recuperação de informação por meio de computadores, e outras mais.

Seja ainda um trabalho sobre a técnica da empunhadura na prova de salto com vara, utilizada por um atleta africano na Olimpíada de Munique. Esta mesma frase não deixa, segundo a definição, de ser uma palavra chave. Não é considerada como tal, entretanto, devido ao tamanho, excessivamente longo: dezoito palavras. Ficam melhor, como palavras chave, as seguintes: empunhadura, salto com vara, atletismo, Olimpíadas, Munique.

No entanto, se este trabalho tivesse sido publicado numa revista de esportes, poderia perfeitamente ter recebido o título "Joaquim Bula-Bula bate sensacional record!", do qual pouco se pode inferir sobre o texto propriamente dito. É necessária, então, a presença de um analista, que possa enriquecer o título, estabelecendo descritores.

E isso pode ser perigoso. No afã de descrever todos os aspectos mencionados pelo autor, para permitir uma futura recuperação mais completa, o indexador poderia ter criado ainda os descritores Alemanha, África, competições internacionais, e até mesmo, o salto com vara na África!

Essa imprecisão, decorrência natural da sistemática de cria

ção de termos chave, pode ser evitada (ou enormemente diminuída), pela adoção de vocabulário controlado, um thesaurus. Infelizmente, este progresso só é possível com um cuidadoso estudo a longo prazo, com pessoal altamente especializado, e com custos assustadores. As palavras chave, embora relativamente imprecisas, oferecem uma solução imediata, de baixo custo, e, apesar de tudo, de razoável efetividade.

O problema das palavras chave, como dissemos acima, é a inexistência de uma forma padronizada de indexação. Assim, um indexador, ao se defrontar com o título "Navegação Aérea", consciente do fato de que interesses diversos poderiam ser satisfeitos com este trabalho, pode descrevê-lo com larga redundância, através das palavras chave aérea, aeronáutica, avião, aeroplano, aeronave, aeroporto, navegação, giroscópio, bússola, rádio-compasso e muitas outras.

Por sua vez, o usuário, conhecendo esta problemática, deve usar, como termos de busca, toda a sinonímia conhecida para o elemento desejado. Assim, um interessado em frio industrial, deve procurar também em refrigeração, ar condicionado, frigoríficos, congelamento, etc.

Apesar destas inconveniências, o sistema de palavras chave pode ser muito útil numa pesquisa bibliográfica. O pior risco que corre o usuário é a recuperação falsa, pela ausência de informação contextual, inerente ao método. É bastante conhecida a estória do usuário que, interessado em ter um aquário, realizou uma busca das palavras "PEIXES" e "ORNAMENTAIS", tendo sido contemplado com o título "Subvenção federal para o desenvolvimento de flores ornamentais, no município de Peixes". Outro risco é a ausência de recuperação, mesmo quando a informação existe. Este problema, porém, é menor, e pode, como já vimos, ser resolvido, tanto por uma elevada redundância na indexação, quanto por uma atitude flexível do usuário em relação aos termos usados em sua busca.

Pretendemos, nestas rápidas considerações, dar uma idéia do

conceito, o suficiente para permitir um entendimento do resto do trabalho, e, mais importante ainda, do uso do sistema criado; finalmente, alertamos nossos futuros usuários quanto a possíveis frustrações.

Queremos, ainda, deixar bem clara nossa intenção ao elaborar esta tese. Nosso interesse é voltado para a melhor utilização dos computadores, e não para o delicado aspecto de como indexar uma publicação. Reconhecemos as dificuldades de criar meios que possibilitem a precisa recuperação de uma referência bibliográfica. Para aqueles que, entretanto, estiverem interessados em uma visão mais ampla deste ramo de conhecimento, recomendamos as referências 1, 2, 8, 9 e 10.

2.3 - NECESSIDADE DE UMA LINGUAGEM DE BUSCA.

Para poder recuperar, de um conjunto de referências bibliográficas, aquelas que possuam elementos comuns, seja o nome do autor, ou o assunto a que se referem, é necessário que possamos enunciar, com precisão, estes elementos comuns, que servirão para a recuperação destas mesmas referências.

Conhecido o objetivo, é possível iniciar um processo sistemático de procura, de busca dos elementos desejados.

A enunciação, sem ambiguidade e com precisão, das características dos elementos que queremos buscar, exige regras bem definidas. Regras desta natureza recebem a denominação de linguagem.

A linguagem de busca aqui apresentada, em particular, e a todo o sistema de recuperação criado, atribuímos o nome SIRIUS - "Sistema de Recuperação de Informação Utilitária Seleccionável".

2.4 - ASPECTOS DA IMPLEMENTAÇÃO.

O usuário fornece os comandos da linguagem de busca através

de cartões perfurados. Estes comandos são analisados do ponto de vista sin
tático, e semântico, e, se livres de erros, imediatamente executados.

A execução do comando corresponde à confecção de uma lista de referências que satisfazem às restrições impostas. Sucessivas expres
sões, ou comandos, aumentam ou diminuem a lista, conforme sejam mais ou menos restritivas. Finalmente, a relação dos trabalhos encontrados é im
pressa.

O sistema é, portanto, interpretativo. O resultado é obti
do diretamente a partir dos comandos-fonte, sem geração de código interme
diário. Isto faz com que o núcleo do sistema seja razoavelmente simples,
a complicação aparecendo apenas durante a montagem do relatório, que é re
lativamente sofisticado

Os programas que formam este sistema de recuperação de in
formação foram escritos para os principais computadores da Divisão de Pro
cessamento de Dados do INPE, e para aplicação imediata. Conseqüentemente,
seus aspectos práticos refletem o estado atual daqueles computadores. Is
to é, as expressões para pesquisa serão codificadas em cartões perfurados
e submetidas em regime de "batch". Entretanto, os programas foram escritos
prevendo a utilização, no futuro, de consultas "on-line" através de termi
nais remotos; para tal serão necessárias modificações de pequena monta nos
atuais programas.

Ainda dentro do espírito de adequação às possibilidades das
máquinas da D.P.D., que apresentam relativamente pequena capacidade de ar
mazenamento em memória secundária, de acesso aleatório, fomos obrigados a
estabelecer algumas restrições adicionais. A principal delas é a necessi
dade de uma preparação dos dados, visando rapidez na recuperação e, ao mes
mo tempo, economia de espaço em disco. Conseqüências imediatas desse pro
cedimento são a necessidade de trabalhar com quantidades pequenas de in
formação - não mais que 15.000 referências - e a impossibilidade de atua

lizar e aumentar esta massa automaticamente: o acréscimo de referências só pode ser feito re preparando-se todos os dados, o que, embora não seja tarefa difícil, é desagradável e dispendioso. Convém lembrar, porém, que esta é uma restrição exclusivamente material, e que pode ser eliminada a qualquer momento.

Lembramos mais uma vez que o sistema é orientado principalmente para referências bibliográficas. Sua utilização em outros dados está condicionada a uma preparação equivalente, e a sua transformação para o formato de referências bibliográficas.

CAPÍTULO III

A LINGUAGEM DE BUSCA

3.1 - INTRODUÇÃO.

Neste capítulo descrevemos todas as características da linguagem de busca, sua sintaxe e sua semântica. Fugimos, entretanto, dos formalismos associados com a sintaxe, e adotamos uma forma explicativa, baseada em grande número de exemplos, que facilitam a sua compreensão.

Preferimos esta maneira, ainda que com riscos de permitir ambiguidades na sintaxe, evitáveis através do uso da forma de Backus-Naur, por exemplo, porque pretendemos que este capítulo seja uma fonte clara para o interessado em usar o sistema, o qual poderia confundir-se com os aspectos altamente formais da BNF (vide apêndice D). Ademais, a própria simplicidade da linguagem torna pequena a probabilidade de deixarmos com interpretação duvidosa algum aspecto da sintaxe. Mesmo nestes casos, esperamos que os exemplos elucidem as dúvidas.

Aspectos práticos, relativos à implementação da linguagem no computador são discutidos no capítulo IV.

3.2 - CONJUNTO DE CARACTERES DISPONÍVEIS.

Todos os caracteres disponíveis nas impressoras dos computadores podem ser utilizados. Alguns deles, porém, estão reservados para simbolizar algumas operações dentro da linguagem.

Arbitrariamente, podemos dividir os 64 caracteres disponíveis nos seguintes grupos:

letras A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

dígitos 0 1 2 3 4 5 6 7 8 9

caracteres especiais
com funções próprias & - / () = * \$ # @ ; " [] espaço < > +

caracteres especiais
sem função ' : + ≤ ≥ ≠ % ? , .

3.3 - COMPONENTES BÁSICOS.

A linguagem apresenta três tipos de componentes básicos: palavras-chave, termos auxiliares e nome de autores. Cada um deles é discutido separadamente: as palavras-chave no capítulo anterior e os demais mais adiante.

Palavras-chave distinguem-se dos demais componentes porque aparecem sempre encerradas entre aspas. Exemplos:

"EDUCACAO" "PESQUISA" "ALCALINO"

Nomes de autores distinguem-se dos demais componentes por que aparecem sempre encerrados pelos símbolos < >, popularmente conhecidos como "parênteses bicudos".

Exemplos: <CUNHA> <ALVES T> <PIMENTEL EGH>

Denota-se um termo auxiliar por um cordão (string) de letras e/ou dígitos, com no mínimo um e no máximo quatro caracteres.

Serviriam como identificador de termos auxiliares os seguintes cordões:

PAO	MESA	J9T3	K	AAAA
18	250T	1Z4	5	4866

Não seriam válidos os seguintes:

SILVA AGRICOLA 123457 T:L 2\$5 A*3

Os três primeiros por terem mais de quatro caracteres e os três últimos por englobarem um caracter não permitido, isto é, nem todos os caracteres são letras ou dígitos.

3.4 - DELIMITADORES.

Delimitadores são símbolos que servem para separar as várias entidades que compõem um programa. À parte esta função, cada delimitador possui um significado especial, que é explicado em outras seções.

São delimitadores os sinais:

= & - / ; () | | espaço < > " +

O símbolo + é usado internamente pelo programa, e, portanto, não pode ser usado em expressões de busca.

3.5 - EXPRESSÕES DE BUSCA

Podemos entender melhor o mecanismo de recuperação de informação, se considerarmos o conjunto de todas as referências como formado pela união de conjuntos menores, formados, por sua vez, pelas referências que contêm um mesmo dado em comum, seja uma palavra-chave ou o nome de um autor. O que desejamos é uma maneira de manipular, ou operar estes conjuntos elementares, impondo restrições mais elaboradas.

Para esclarecer melhor o processo, suponhamos duas pala

avras-chave quaisquer. Para cada uma delas há um conjunto de referências que as admitem como palavras-chave. Podemos agora, a partir destes dois conjuntos, formar um terceiro. Por exemplo, o terceiro conjunto pode conter apenas as referências comuns aos dois conjuntos iniciais; ou então, arbitrariamente escolhemos certo número de elementos de cada um, formando com eles o terceiro conjunto.

Chamamos de operação de busca, ou simplesmente de operação, a regra de formação de um conjunto a partir de dois outros.

As operações de busca são, portanto binárias: exigem a presença de dois operandos, no caso os conjuntos.

A forma codificada de indicar uma sequência de operações é chamada de expressão de busca. Nela, as operações são indicadas por um símbolo convencional, chamado operador, colocado entre os operandos.

O valor de uma expressão de busca é o resultado final da aplicação das operações nela indicadas, ou seja, é um conjunto de referências.

O valor de uma expressão de busca pode ser um conjunto vazio.

3.5 - OPERAÇÕES DE BUSCA.

O exemplo mais simples de expressão de busca seria uma única palavra-chave; por exemplo:

"IONOSFERA"

(1)

Se a expressão de busca (1) fosse empregada para a recuperação, obteríamos todos os artigos que têm "ionosfera" como palavra-chave. Esta relação poderia ser extremamente volumosa.

Podemos restringir a procura das referências adicionando outra palavra-chave à nossa expressão de busca, de modo a recuperar apenas as referências que possuam as duas palavras-chave como descritores.

Fazemos isso assim:

"IONOSFERA" E "FOGUETES"

(2)

Satisfaria a expressão (2) o artigo "Uso de foguetes de sondagem para o estudo da ionosfera", que possui os dois descritores, ionosfera e foguetes.

Outro operador disponível é a palavra OU.

Além das referências citadas anteriormente, satisfariam a expressão:

"IONOSFERA" OU "FOGUETES"

(3)

as referências "Curvas de empuxo de foguetes Nike-Apache" e "Uso de foguetes pirotécnicos na comemoração da independência" e até mesmo "Japoneses ganham a competição nadando como foguetes".

O conetivo OU pede a existência de uma das palavras-chave, independentemente da ocorrência da outra.

Finalmente, MAS NAO é o terceiro operador permitido. Tem sentido restritivo, tal como o conetivo E. No entanto, enquanto este exi

ge a presença dos dois operandos, o operador MAS NAO pede a presença do primeiro e a ausência do segundo. Vejamos um exemplo:

"MINERACAO" MAS NAO "FERRO" (4)

"FERRO" MAS NAO "MINERACAO" (5)

É evidente que um trabalho do tipo "Mineração de ferro no Pará" é eliminado tanto por (4) quanto por (5). Mas as idéias por trás das expressões são bem diferentes. No caso (4), estamos interessados em mineração, e no caso (5), em ferro.

Suponhamos os títulos "Técnicas de mineração", "Mineração de prata", "Ligas de ferro" e "Tipos de ferro para construção". Os dois primeiros satisfazem a expressão (4) e os dois últimos a expressão (5). É interessante notar que "FERRO" e "MINERAÇÃO" recusaria os quatro, e que "FERRO" ou "MINERAÇÃO" aproveitaria os quatro.

As operações E e OU são comutativas e associativas, isto é,

A E B = B E A
A OU B = B OU A

comutatividade

(A E B) E C = A E (B E C)
(A OU B) OU C = A OU (B OU C)

associatividade

Devido à característica de associatividade, sequências de operadores E e OU podem ser escritas sem necessidade de parenteses, como em

A E B E C E D
A OU B OU C OU D

Já a operação MAS NAO não apresenta nenhuma dessas características, como é mostrado no apêndice E. Ou seja,

A MAS NAO B ≠ B MAS NAO A

(A MAS NAO B) MAS NAO C ≠ A MAS NAO (B MAS NAO C)

Descrevendo a ação dos operadores E, OU e MAS NAO, explicados acima, em termos de conjunto de referências temos: a operação E forma um terceiro conjunto no qual figuram apenas os elementos que existem, ao mesmo tempo, nos dois iniciais. Em suma, é a intersecção dos conjuntos. A operação OU forma um terceiro conjunto no qual figuram todos os elementos que existem tanto num quanto noutro conjunto inicial. Em suma, é a união dos conjuntos. A operação MAS NAO forma um terceiro conjunto no qual figuram os elementos do primeiro conjunto que não figuram no segundo. Em suma, é a diferença dos dois conjuntos.

3.7 - SÍMBOLOS ALTERNATIVOS PARA OS OPERADORES.

Os operadores podem ser denotados, nas expressões de busca, tanto por sua forma por extenso quanto por uma forma convencional. Esta última corresponde aos seguintes sinais:

E	&
OU	/
MAS NAO	-

As duas formas são válidas, podendo ser usadas indistintamente, numa mesma expressão.

Por exemplo, está correto dizer

"AZUL" OU "VERDE" / "ROSA"

3.8 - PRECEDÊNCIA DOS OPERADORES.

É permitido o uso, em uma expressão, de operadores diferenciados

tes. Por exemplo, é válida a expressão

$$A \text{ OU } B \text{ E } C \quad (7)$$

onde A, B e C representam, genericamente, palavras-chave.

A expressão (7) pode ser interpretada de duas maneiras distintas:

$$A \text{ OU } (B \text{ E } C) \quad (8)$$

$$(A \text{ OU } B) \text{ E } C \quad (9)$$

com significados distintos. Em (8) queremos os títulos que tratem de A ou tratem de B e C, simultaneamente; ao passo que em (9) queremos títulos que contenham A e C, simultaneamente, ou então que contenham B e C, simultaneamente.

Resolve-se esta ambiguidade estabelecendo relações de precedência entre os operadores. Isto é, uma expressão é avaliada realizando primeiro as operações que envolvam operadores de maior precedência.

A relação de precedência para nossa linguagem de busca é:



Operações dentro de uma expressão são executadas na ordem de precedência, da maior para a menor. Quando, na expressão, ocorrem operadores de mesma precedência, a expressão é calculada da esquerda para a direita.

Quando for necessário executar as operações em ordem diferente daquela ditada pela precedência dos operadores, lançamos mão de pa

rênteses ou colchetes. Quando uma expressão é colocada entre parênteses ou entre colchetes, ela é considerada um operando, ou seja, as operações entre parênteses são efetuadas antes das demais, independentemente da precedência dos operadores envolvidos.

É permitida a criação, em uma expressão entre parênteses, de sub-expressões também entre parênteses, em qualquer grau de profundidade.

Os pares de símbolos ([e]) são absolutamente equivalentes, podendo, portanto, ser usados indistintamente.

Desta forma, a expressão (7) significa realmente

A OU (B E C)

3.9 - OPERANDOS DE BUSCA.

Mencionamos, na discussão das expressões de busca, que operandos são conjuntos formados a partir dos sub-conjuntos iniciais que compõem o data-base. Na discussão anterior, todos os operandos apresentados eram palavras-chave, o que pode ter deixado a impressão de que palavras-chave são os únicos operandos possíveis, o que é falso. Temos dois outros tipos de operandos: nomes de autores e termos auxiliares.

3.9.1 - Nomes de autores.

Conjuntos iniciais, equivalentes aqueles das palavras-chave, existem também para os autores

Palavras chave, porém, são simplesmente palavras, ou, na pior das hipóteses, uma frase. Designar uma palavra é simples; basta colocá-la entre aspas.

Em se tratando de nomes próprios, a coisa se complica. Há prenomes e sobrenomes, em quantidades variadas. A publicação do IBBD, que regulamenta a referência bibliográfica para o SIABE (referência 14), indica o seguinte procedimento: " ... a entrada é feita sempre pelo último sobrenome, seguido das iniciais.

Exemplo: Souza, E.F. de M."

Ora, seria extremamente difícil recuperar este autor procurando uma citação como o exemplo acima, com vírgulas, pontos e preposições. Qualquer distração por parte do usuário, como por exemplo, a ausência de uma vírgula ou de um ponto poderia significar a perda de recuperação. Adotamos, então, um critério de simplificação, descrito com detalhes no apêndice A. Aqui, mencionamos apenas os aspectos principais do critério, e suas implicações na recuperação.

O autor acima é cadastrado como SOUZAbEFM, onde o símbolo b significa um espaço em branco. Para recuperar este autor, pode ser usado <SOUZA>, <SOUZAbE>, <SOUZAbEF> ou <SOUZAbEFM>. O quadro abaixo clarifica melhor a idéia de recuperação do nome do autor. Como já foi dito anteriormente, o nome do autor, em qualquer das formas escolhidas, deve ser delimitado pelos "parênteses bicudos", ou seja, os caracteres < e >.

	Recupera	Não recupera
<SOUZA>	SOUZA A SOUZA SRM SOUZA H	
<SOUZA E>	SOUZA E SOUZA EF SOUZA EX	SOUZA CR SOUZA M
<SOUZA EA>	SOUZA EA SOUZA EAL	SOUZA ET SOUZA E SOUZA TR

3.9.2 - Termos auxiliares.

Uma comodidade muito interessante é a habilidade de dar um nome simbólico a um conjunto qualquer, e posteriormente usar este nome a cada referência ao conjunto, sem necessidade de reescrever a expressão de busca que deu origem ao conjunto.

Esta comodidade é provida, na linguagem de busca, pelos termos auxiliares. Qualquer conjunto pode receber um nome, ou seja, um termo auxiliar pode ser criado para o representar, quer o conjunto provenha de uma palavra chave, do nome de um autor ou de uma expressão de busca.

Uma vez definido, um termo auxiliar assume os direitos de um operando, podendo, portanto, ser usado em novas expressões de busca. No entanto, dada a associação bi-unívoca entre o identificador usado e seu significado, o identificador não pode ser redefinido.

A associação entre o termo e a expressão se faz escrevendo o identificador, o sinal = e a expressão.

Exemplos a seguir:

A = "ATMOSFERA"
B = "CARROS" E "ELETRICOS"
JM = <MESQUITA J>
INP = "INSTITUTO" & "PESQUISAS" & "ESPACIAIS"
INPE = "INPE"/INP

Nos dois últimos exemplos, vemos uma aplicação do conceito. Definimos o termo INP como significando a ocorrência simultânea das palavras-chave Instituto, Pesquisas e Espaciais. Claramente, esta definição de tetará menções ao INPE onde o nome é escrito por extenso. Como também de sejam recuperar as referências que usem a abreviatura, fizemos nova de finição, dessa vez o termo INPE, que reúne a palavra-chave "INPE" e o térmo

INP, através do operador OU. Temos agora a certeza de detetar qualquer referência ao Instituto, seja através do nome completo, seja através da sigla.

3.10 - PREFIXOS COMUNS.

Uma dificuldade que o usuário poderia encontrar em um processo de recuperação decorre do fato de que nem sempre se sabe exatamente a forma pela qual a idéia que perseguimos terá sido registrada. Eventualmente, poderemos encontrar o tópico correlacionado com muitas formas diferentes da mesma palavra.

Suponhamos uma pesquisa em torno da idéia "Universidade". É muito viável que diferentes autores tenham entitulado trabalhos que nos interessariam usando as palavras universidade, universidades, universitário, universitários, université, university e outras.

Seria extremamente desagradável ter de codificar todas estas palavras-chave e ligá-las através do operador OU. E, realmente, dispõe a linguagem de busca de um recurso. Codificamos, apenas, como palavra-chave, "UNIVERSI*".

O uso do asterisco como último caracter de uma palavra-chave tem o sentido de uma múltipla operação OU. O conjunto formado é a união dos conjuntos elementares de todas as palavras-chave que têm, como raiz, os caracteres que antecederam o * na palavra-chave.

O asterisco pode ser empregado também para nome de autores. Assim, <CARV*> recuperará todos os Carvalhos, Carvalhaes, etc.

CAPÍTULO IV

A CODIFICAÇÃO DOS PROGRAMAS DE BUSCA

4.1 - PROGRAMAS DE BUSCA.

Programas de busca são a materialização de comandos da linguagem de busca, explicados de forma teórica no capítulo anterior. Agora, discutiremos aspectos práticos da pesquisa bibliográfica, enfatizando a maneira correta de ajustar os comandos de busca aos cartões perfurados.

Existem três tipos de construções, ou comandos: identificação do usuário, definição de termos de busca e bibliografias procuradas. São também possíveis, comentários.

Estes comandos são escritos em um formato algo limitado. Na implementação da linguagem, optamos pela facilidade de interpretação dos comandos, mesmo a custo de perda de flexibilidade de programação. As consequências principais dessa opção são: necessidade de identificação do tipo do cartão, feita pela presença de um caracter especial, arbitrariamente escolhido, numa posição bem determinada do cartão, e obrigatoriedade de se colocar todo o comando em um mesmo cartão. Esta última restrição pode significar a necessidade de criação de termos auxiliares e seu uso posterior.

4.2 - CAMPOS DO CARTÃO-FONTE.

coluna 1 - um único caracter que identifica o tipo de frase. São válidos:

\$ - identificação do usuário.

- definição de termos de busca auxiliares.

@ - expressão de busca, definindo uma classe de referências desejadas, usando ou não termos auxiliares.

- comentários - o caracter na coluna 1 é um espaço em branco.

colunas 2 a 72, inclusive - a frase da linguagem integralmente perfurada neste espaço. Não são permitidos cartões de continuação, sendo responsabilidade do usuário dividir a frase ou comando de busca em sub-comandos, caso necessário.

colunas 73 a 80 - número de sequência dos cartões. Opcionalmente, pode-se numerar os cartões, para conservar a integridade da massa de cartões.

4.3 - DESCRIÇÃO DETALHADA DOS DIVERSOS COMANDOS.

4.3.1 - Comentários.

Pode ser usada como comentário qualquer sequência de caracteres disponíveis nas impressoras.

Os comentários podem aparecer em qualquer número; o significado do comentário propriamente dito é completamente ignorado pelo interpretador, não influenciando na busca.

4.3.2 - Cartão de identificação (\$ na coluna 1)

Neste cartão, único em todo o programa, são colocados alguns dados pessoais do usuário.

Estes dados, todos opcionais, de qualquer comprimento, são: nome do usuário, título da pesquisa e endereço. Tais informações serão produzidas no relatório, e ajudarão na devolução da pesquisa.

Os dados do usuário são codificados em formato livre, se parados por vírgulas. É possível omitir qualquer dos dados, como se pode perceber pelos exemplos abaixo:

```
$ I.L.R.Teixeira, Referências sobre alumínio, INPE
$ , Busca de informações sobre florestas tropicais
$ João Aurora, Rua Guilherme Florencio 84
$ Batista Pereira
$ ,, Projeto SERE
```

4.3.3 - Definição de termos auxiliares (# na coluna 1)

A forma de um comando de definição de termo auxiliar é: o identificador do termo auxiliar, o sinal = e a expressão de busca, sendo possível o uso, na expressão de busca, de palavras-chave, nomes de autor e termos auxiliares já definidos, interligados pelos operadores (e parênteses, eventualmente), como já discutido na Seção 3.9.2.

4.3.4 - Expressão desejada (e na coluna 1)

O cartão e é usado para indicar uma expressão de busca da qual queremos a listagem do conjunto de referências a ela associado.

Hã duas formas para a sua codificação: uma expressão de busca, simplesmente, ou um identificador de termo auxiliar, o sinal = e uma expressão de busca.

No primeiro caso, as referências que preenchem as condições da expressão são impressas, e o programa continua. No segundo caso, além da impressão, o comando funciona como uma definição de termo, o qual podêrá ser usado mais adiante. Vejamos um exemplo:

```
LINE = "TELE*" ; LINE - "TELEFONE"
```

Temos, inicialmente, a listagem dos termos que satisfazem a "TELE*"; adicionalmente, a este conjunto damos o nome de LINE. No segundo comando, colocado no mesmo cartão, pedimos a impressão do subconjunto de LINE que não contem a palavra-chave "TELEFONE".

4.4 - OUTRAS CONSIDERAÇÕES SOBRE AS EXPRESSÕES DE BUSCA.

Um termo auxiliar, de qualquer tipo, s̄o pode intervir numa express̄o de busca, como operando, depois de ter sido definido, isto ẽ, de ter aparecido ˆ esquerda do sinal =, num comando anterior.

Dentro do campo destinado ao comando, ou seja, entre as co lunas 2 e 72, inclusive, o comando pode ser escrito em formato livre, com a exceç̄o das iniciais do autor, que, quando ocorrerem, devem estar sepa radas do sobrenome por um ˆnico branco.

Como os operadores =, &, - e / s̄o tambẽm delimitadores, qualquer nˆmero de brancos pode ocorrer entre eles e seus operandos, co mo ẽ indicado abaixo, onde aparecem definiç̄es de termos corretas.

```
A = B&C
E = G & "RIO"
EB = <BRAGA E>
```

Entretanto, quando se usar a forma nˆo simbˆlica dos ope radores, eles devem ser separados dos operandos por pelo menos um espaço em branco. Naturalmente, tanto esta forma dos operadores quanto os iden tificadores de termos s̄o entendidos como palavras, e, portanto, entre seus caracteres nˆo deve haver branco. Vide exemplos a seguir.

```
A = B OU C           (correto)
A=B OUC             (errado - falta espaço depois do OU)
A T = A&B           (errado - AT deve aparecer junto)
```

C = A O U B (errado - OU deve aparecer junto)
D = A MAS NAO B (correto)

A coluna 73 do cartão é um delimitador natural, visto que o comando deve terminar antes dela. A coluna 72, porém, é perfeitamente usável.

Não é necessário delimitar o fim do comando; este é percebido pela ausência de caracteres adicionais. No entanto, é possível aproveitar um mesmo cartão para mais de um comando: estes são separados por um ponto-e-vírgula.

Exemplos:

A = B&C ; D = A/G ; H = A&B&D

Quando se usam, num mesmo cartão, diversas expressões de busca, aquelas mais a direita podem usar, como operandos, termos definidos mais a esquerda.

4.5 - ESTRUTURA DE UM PROGRAMA.

Em uma única execução do interpretador, podem ser submetidas diversas pesquisas bibliográficas, de uma mesma pessoa ou de pessoas diferentes.

O domínio de uma pesquisa, ou de um programa de busca, é determinado pelo cartão de identificação. Este cartão é, portanto, imprescindível e único, encabeçando os demais cartões relativos aos termos.

Depois do cartão \$, podem aparecer, em qualquer ordem e em qualquer número, os cartões #, @ e comentário.

Não existe uma indicação explícita do fim. Este é determinado, ou pelo aparecimento de outro cartão \$, identificando um novo pro

grama de busca, ou pelo fim físico da massa de cartões. Conquanto não indicada explicitamente, esta fronteira existe, e cumpre um papel muito importante: determina os limites de validade das definições.

É completa a liberdade do usuário na escolha dos identificadores para os termos que ele queira definir. É muito possível, por isso mesmo, que duas pesquisas usem o mesmo identificador com significados diferentes, e que ambos os programas sejam submetidos numa mesma corrida do interpretador. Mostremos, num exemplo, esta possibilidade.

```
$ JOAO
:
:
:
HOJE = "TEMPO"/"ATUAL"
:
:
:
$ JOSE
:
:
:
HOJE = "DIA" OU "MES"
:
:
```

Dissemos anteriormente, que um indicador só podia aparecer à esquerda do sinal =, isto é, ser definido, uma única vez. No exemplo acima, o termo HOJE é definido duas vezes. No entanto, isto não constitui infração das regras, porque, entre as duas definições, existe um cartão \$.

Ou seja, a definição usada por JOAO não conflita com a de JOSE, porque pertence a programas de busca diferentes. Dizendo a mesma coisa com outras palavras: as definições de JOAO têm valor apenas para a pesquisa de JOAO. O aparecimento do cartão \$, iniciando a pesquisa de JOSE, e, naturalmente, terminando a de JOAO, anulou todas as definições já feitas, possibilitando a sua redefinição.

Em termos práticos, podemos dizer que o encerramento de uma pesquisa, por intermédio do início de outra, destrói todos os conjuntos, ou listas, de referências já criadas, devolvendo as referências ao conjunto original.

CAPÍTULO V

ALGORITMOS DE RECUPERAÇÃO

5.1 - INTRODUÇÃO.

Este capítulo é dedicado ao estudo dos algoritmos que recuperam as informações, e que são invocados pelo interpretador, a medida que este analisa o programa de busca.

Os algoritmos baseiam-se na capacidade dos dispositivos de armazenagem do computador de sofrerem acesso aleatório; desta maneira, qualquer dado pode ser obtido dentro de um período de tempo constante, independentemente da sua posição dentro do arquivo ou de seu conteúdo.

Não consideramos a criação desses arquivos como parte integrante do nosso sistema. É claro, entretanto, que essa criação é uma etapa muito importante, pois dela depende o sucesso da recuperação. Descrevemos, porém, com bastante detalhe, o aspecto final que os arquivos adquirem após sua criação. Este aspecto é principalmente voltado para as características de referências bibliográficas. Outros dados podem, porém, ser convertidos a este formato, nisto consistindo o esforço de adaptação, ao nosso sistema, de outros tipos de informação.

Cumpre ressaltar aqui, que o processo de recuperação é capaz de reconhecer duas classes distintas de chaves de recuperação, quais sejam, autores e palavras-chave, as quais, embora semelhantes, sob o ponto de vista do programador, são completamente distintas para o processo, estando, inclusive, estabelecidas em arquivos separados.

Mais uma vez lembramos que o objetivo primordial deste trabalho é a recuperação de referências bibliográficas, para as quais sempre temos autores e descritores. Como nos limitamos, propositadamente, a apenas duas classes, consideramos eficiente o método de identificá-las por um caracter especial. Houvesse uma terceira classe, digamos, a imprensa, teríamos ainda, além das aspas e dos parênteses bicudos, um outro caracter que caracterizasse esta terceira classe, quem sabe, apóstrofes.

Mas não é a forma o principal, e sim o fato de que o processo aqui apresentado é capaz de lidar com mais de uma classe de termos. A maneira de diferenciá-los vai depender da quantidade de classes. Quando estas são em pequeno número, o encerramento do termo por um caracter especial é prático. Para número maior de classes, podemos apelar para métodos mais explícitos, como designar a classe juntamente com o termo.

5.2 - A REFERÊNCIA BIBLIOGRÁFICA TÍPICA.

Uma referência bibliográfica indica, em geral, o autor ou autores, o título do trabalho e dados sobre a sua publicação.

Por exemplo:

Rabello, E.

Pesquisas e perspectivas da indústria farmacêutica
R. Bras. Farm. 51(2)85-92, MAR/ABR 1970

Bier, O.

Bacteriologia e Imunologia em suas aplicações a medicina e
a higiene - 14.^a edição, rev. e ampl.
São Paulo, Melhoramentos, 1970. XIV., 983 p.

A citação da referência deve compreender todos os dados. A través deles o interessado na pesquisa ajusta o seu interesse, e, para os trabalhos realmente fundamentais, busca a publicação que o contém.

5.3 - AS REFERÊNCIAS BIBLIOGRÁFICAS EM DISCO.

Adotamos, para este arquivo, um formato essencialmente econômico em espaço.

Forma-se, com os dados da referência, um cordão de caracteres, onde cada dado é separado do seguinte por um caracter especial, que age como delimitador. Uma referência é transformada, então, em uma sequência de campos, separados uns dos outros por uma marca especial. Cada campo é do menor tamanho possível, eliminando-se brancos a direita.

Neste processo a referência:

Gama, G. G. da
Febre Reumática
R. Assoc. Med. M. Gerais 21(1)43-53, Mar. 1970

fica reduzida a

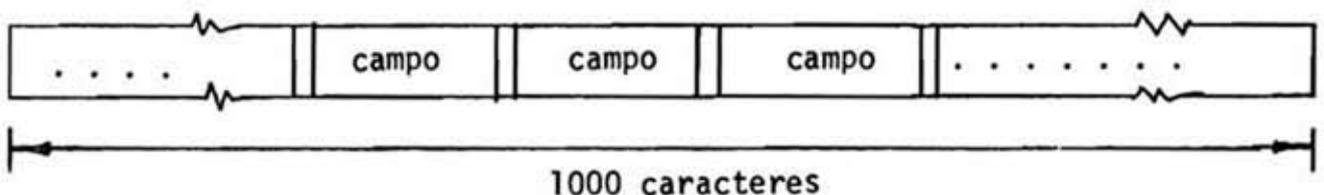
Gama, G. G. da*Febre Reumática*R. Assoc. Med. M. ...

↖ separadores ↗

Quando da impressão, os diversos dados são expandidos, apresentando-se a referência na sua forma padrão.

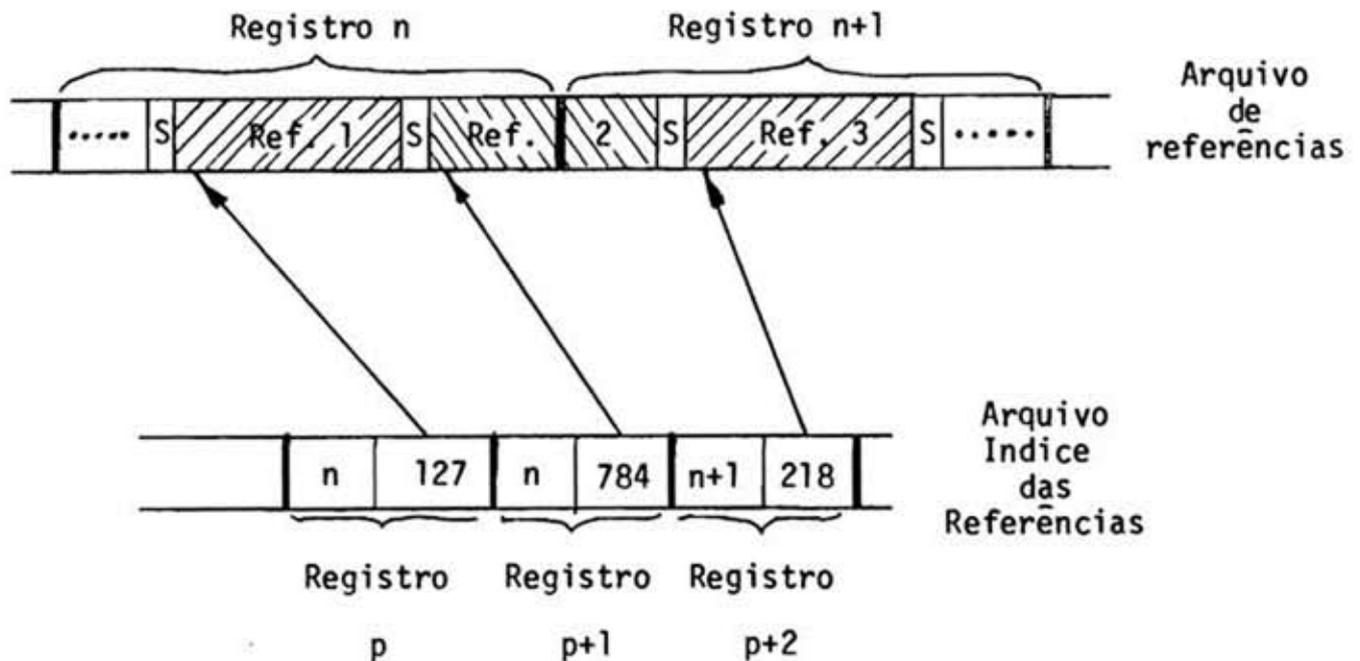
Utilizando-se este processo, cada referência é transformada num cordão de comprimento variável. No fim de cada cordão é colocado um segundo caracter especial, e diversos deles são arranjados desta forma até completar 1.000 caracteres, quando então são gravados.

Esquematicamente, o registro do arquivo tem a seguinte apresentação:



o intervalo entre campo é um separador.

Num arquivo a parte, registramos, para cada referência, o número ou chave do registro onde ela começa, e a quantos caracteres do começo do registro aparece o primeiro caracter da referência. A este arquivo chamamos de "Índice das referências".



É elementar, a partir do Índice, obter a referência. Fornecemos ao sistema operacional o primeiro dado do registro Índice, ou seja, a chave de acesso aleatório das referências, e o sistema operacional nos devolve um cordão de mil caracteres, no qual temos, pelo menos, um pedaço da referência completa.

Dentro desses mil caracteres, podem existir várias referências. Não há ambiguidade, porém, porque conhecemos exatamente a posição, dentro dos mil, onde começa a referência desejada; este dado é fornecido pelo segundo campo do registro Índice.

Portanto, é muito simples a obtenção da referência a partir do Índice. Mas, e o Índice? No fundo transferimos o problema de achar

uma referência para achar o índice dessa referência. E isso, como se faz?

Se repararmos que a criação do arquivo das referências é concomitante com a criação de seu índice, veremos imediatamente que há uma relação direta entre a chave de acesso aleatório ao índice e a posição que a referência ocupa em relação as demais. Assim, para conhecer os dados sobre a milésima referência, basta pedir ao sistema operacional o registro de chave 1000 no índice e, a partir deste, encontramos a referência.

Por outro lado, nunca nos interessaremos diretamente por uma referência, a ponto de sabermos seu número de ordem dentro do conjunto, mas sim chegaremos a ela através de seus atributos. Ora, quando criamos os arquivos, sabemos, para uma dada referência, quais são seus atributos. É muito simples, então, manter uma numeração sequencial, de 1 a n, para as referências, e associar, indelevelmente, este número sequencial aos seus atributos. No futuro, quando encontrarmos um atributo, este nos dará o número de ordem da referência, o qual é a chave para o índice, e, finalmente, para a própria referência.

5.4. ARQUIVO DE PALAVRAS-CHAVE.

Durante o estágio anterior, separam-se, para cada referência, os descritores e os autores, associados ao número de ordem da referência. Naturalmente, os elementos assim produzidos são em uma ordem caótica, imprópria para o uso. Faz-se, então, uma classificação, de forma a dispor os descritores e os autores, em ordem alfabética. Essa classificação tem ainda a propriedade de aglutinar as diversas menções a um mesmo descritor (ou autor).

Monta-se, então, para cada descritor, uma lista de números de ordem de referências que admitem aquele descritor. Temos o seguinte "lay-out":

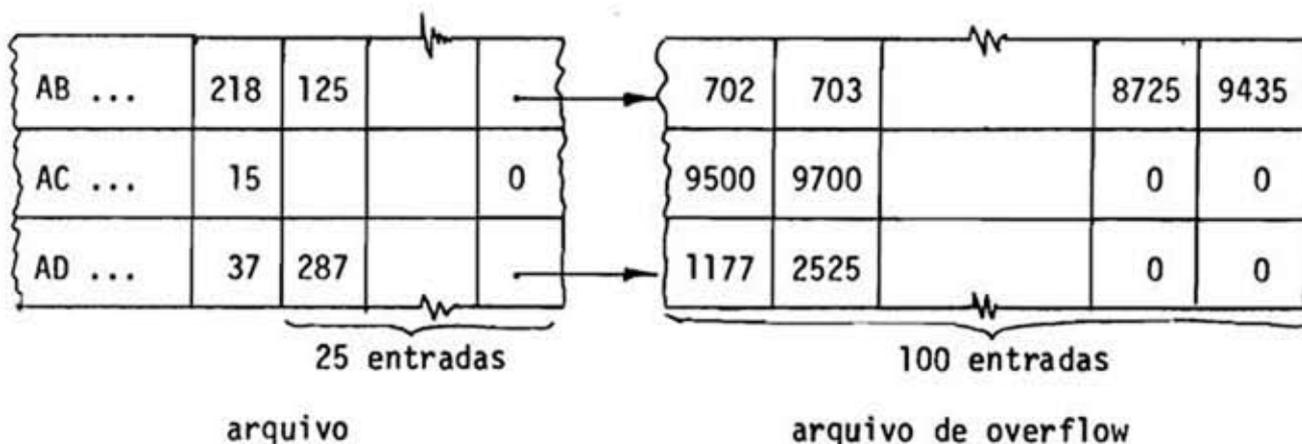
descriptor	nº de referências afetadas	lista de números de ordem
------------	----------------------------	---------------------------

SAÚDE	38	15	27	387		10284
-------	----	----	----	-----	--	-------

Cada entrada nesta tabela, ou cada registro neste arquivo guarda o descritor propriamente dito, quantas referências admitem esta descrição e o número de ordem de cada uma das referências. Assim, conhecido o descritor e seu registro, temos imediatamente caracterizado o conjunto de referências que comportam essa descrição.

Há descritores que geram conjuntos imensos, como por exemplo, "MEDICINA", que facilmente é atribuível a 5.000 ou mais referências. A média, entretanto, é muito mais baixa, da ordem de 10 a 20. Por outro lado, é muito mais simples e prático trabalhar com arquivos onde todos os registros são do mesmo tamanho. Se adotarmos secamente esta técnica, teremos que definir registros com capacidade para cinco mil ou mais referências, e isto seria um desperdício incompatível com as limitações de espaço em disco com que temos de conviver.

Padronizamos, então, o número de referências em 25 por registro. Quando um descritor "tem" mais de 25 referências, mantemos as demais num arquivo a parte, de "overflow". O esquema abaixo ilustra a situação.



Quando um descritor tem mais de 25 referências, a 25ª não é um número de ordem, mas sim um ponteiro para a tabela de overflow, ou seja, a chave do registro onde a lista continua. Cada registro de overflow tem capacidade para 100 referências. Se um registro de overflow não for suficiente, a lista continua em registros sucessivos.

5.5 - PESQUISA BINÁRIA.

Para uma sequência qualquer de elementos, a verificação da ocorrência ou não de um dado elemento na sequência implica na comparação de todos os elementos da sequência com o procurado, até que achemos uma igualdade ou que a sequência se esgote. É fácil concluir que o número médio de testes é $n/2$, onde n é o número de elementos da sequência.

O número de testes pode ser drasticamente reduzido, se os elementos da sequência guardarem entre si uma relação sempre constante. Suponhamos que, para quaisquer dois elementos da sequência, a_i e a_j , com $1 \leq i < j \leq n$, temos sempre $a_i < a_j$ ou $a_i > a_j$, ou seja, a sequência é monotonamente crescente ou decrescente. Conhecida a relação entre um elemento da sequência a_k , digamos, e o termo procurado de valor m , com $a_k \neq m$ podemos dizer imediatamente, que, se houver na sequência um elemento $a_j = m$, ele estará acima ou abaixo de k ; por exemplo, se a tabela for crescente, teremos:

Se $a_k > m$ e $a_j = m$, então $j < k$;

Se $a_k < m$ e $a_j = m$, então $j > k$;

A tabela de descritores, discutida na seção precedente, está organizada em ordem alfabética; o que significa que seus elementos são monotonamente crescentes. Podemos, então, aplicar a esta tabela o processo conhecido como pesquisa binária.

Na pesquisa binária dividimos a sequência a pesquisar em duas outras, de tamanhos os mais aproximados possíveis. Esta divisão é fei

ta calculando o elemento "médio" da tabela, o elemento com índice $i = \left\lfloor \frac{1+n}{2} \right\rfloor$ onde n é o número total de elementos na sequência. Agora, por simples inspeção, descobrimos em qual das duas sequências $a_1 \dots a_i$ e $a_{i+1} \dots a_n$ o elemento procurado está. Em casos felizes, é o próprio elemento a_i . O processo é aplicado sucessivamente às sequências "ganhadoras", até que o elemento seja encontrado ou que a sequência se reduza a um único elemento.

A pesquisa binária exige um máximo de $\lceil \log_2 n \rceil$ testes; qualquer elemento de uma tabela de mil entradas é encontrado depois de não mais de dez comparações. Uma pesquisa sequencial sobre a mesma tabela, faz, em média, 500 comparações.

5.6 - ALGORITMO PARA PESQUISA BINÁRIA.

Seguimos a forma descrita na referência 16 para a descrição de algoritmos.

Algoritmo B (Pesquisa binária em tabelas). Este algoritmo verifica, numa tabela de n entradas, a existência de um termo T , e, se o termo existe, indica o número de ordem da entrada.

B1. [Inicialização] seja BAIXO $\leftarrow 1$ e ALTO $\leftarrow n$;

B2. [Calculo do índice médio] MEIO $\leftarrow \left\lfloor (ALTO + BAIXO) / 2 \right\rfloor$;

B3. [Teste] se ELEMENTO (MEIO) = T então o termo procurado ocorre na tabela na entrada de ordem MEIO. O algoritmo termina;

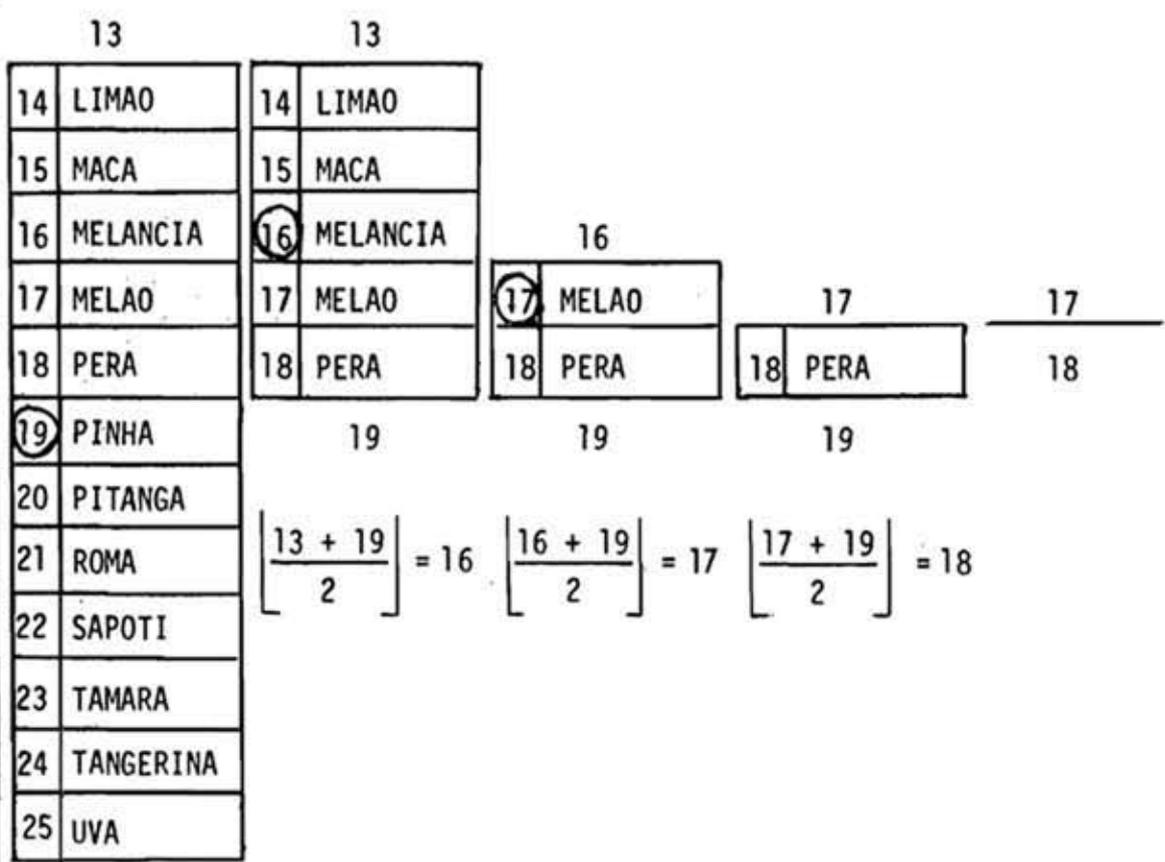
B4. Se ITEM (MEIO) < T então BAIXO \leftarrow MEIO senão ALTO \leftarrow MEIO ;

B5. Se ALTO - BAIXO > 1 então vá para B2. Senão o termo não existe e o algoritmo termina.

MECANISMO DA PESQUISA BINÁRIA

1	ABACATE
2	ABACAXI
3	BANANA
4	CAJU
5	CARAMBOLA
6	CEREJA
7	DAMASCO
8	FIGO
9	FRAMBOESA
10	GOIABA
11	JABOTICABA
12	JACA
13	LARANJA
14	LIMAO
15	MACA
16	MELANCIA
17	MELAO
18	PERA
19	PINHA
20	PITANGA
21	ROMA
22	SAPOTI
23	TAMARA
24	TANGERINA
25	UVA

Passos na busca da entrada MORANGO.



$$\left\lfloor \frac{1 + 25}{2} \right\rfloor = 13$$

$$\left\lfloor \frac{13 + 25}{2} \right\rfloor = 19$$

Figura 1

A figura 1 ilustra a pesquisa da palavra MORANGO, que não existe na tabela.

O primeiro elemento médio achado foi o de ordem 13, LARANJA. Como MORANGO é alfabeticamente maior que LARANJA, seu número de ordem será necessariamente maior que 13.

O segundo teste foi feito sobre o elemento 19, PINHA, maior que MORANGO.

A terceira sequência, dos elementos 13 a 19, tem MELANCIA como meio, item 16.

A quarta sequência, entre 16 e 19, inclusive, tem apenas dois elementos não testados. O meio é o termo 17, MELAO, ainda diferente de MORANGO.

Finalmente, chegamos a quinta sequência, de três elementos, MELAO, PERA e PINHA, itens 17, 18 e 19. Seu elemento médio é o 18, PERA que também não é MORANGO. É importante notar que os três itens presentes nesta sequência já foram testados, e descobriu-se que MORANGO, se existisse, deveria ter número de ordem maior que 17 (MELAO) e menor que 19 (PINHA). A última esperança é 18, que também falha. O critério de subdivisão das sequências conduz, agora, à sequência (17, 18), a qual é vazia. O algoritmo, portanto, termina sem ter encontrado o elemento procurado.

5.7 - ALGORITMOS PARA AS OPERAÇÕES OU, E e MAS NAO.

Tendo por base os arquivos de descritores (ou autores), é possível criar listas de referências. Nestas listas, conhecemos o número de referências que as compõem, bem como sabemos que o número de ordem das referências aparece sempre em ordem crescente.

Seguem-se os algoritmos que efetuam a união, a intersecção

e a diferença de duas listas, nenhuma das quais é vazia.

A notação usada nestes algoritmos é a seguinte:

- as listas chamam-se A e B, respectivamente, e a operação em questão é A/B, A&B ou A-B;
- os elementos das listas são referenciados em forma de subscripto; assim, o terceiro elemento da lista A é indicado por A(3). É possível referência a um termo genérico, por exemplo B(I+2);
- a lista resultado é chamada C, seus elementos sendo designados de forma análoga aos elementos de A e B.

5.7.1 - Algoritmo para a operação OU.

Algoritmo 0. Faz a união de duas listas.

01. [Inicialização] $P \leftarrow Q \leftarrow 1$; $H \leftarrow R \leftarrow 0$; $M \leftarrow$ tamanho de A ;
 $N \leftarrow$ tamanho de B ;
(P,Q,R percorrerão, respectivamente, as listas A, B e C. H é o tamanho da lista C).
02. Se $A(P) < B(Q)$ então ir para 03 ;
Se $A(P) > B(Q)$ então ir para 04 ;
Se $A(P) = B(Q)$ então $Q \leftarrow Q+1$, $N \leftarrow N-1$;
Se $N = 0$ então ir para 05 ;
03. [Elemento de A] $H \leftarrow H+1$, $R \leftarrow R+1$, $C(R) \leftarrow A(P)$, $P \leftarrow P+1$, $M \leftarrow M-1$;
Se $M = 0$ então ir para 06 senão ir para 02 ;
04. [Elemento de B] $H \leftarrow H+1$, $R \leftarrow R+1$, $C(R) \leftarrow B(Q)$; $Q \leftarrow Q+1$, $N \leftarrow N-1$;
Se $N = 0$ então ir para 05 senão ir para 02 ;

05. [Copia resto da lista A] Se $M = 0$, o algoritmo termina;
 $H \leftarrow H+1$, $R \leftarrow R+1$, $C(R) \leftarrow A(P)$, $P \leftarrow P+1$, $M \leftarrow M-1$ e repetir 05;
06. [Copia resto da lista B] Se $N = 0$, o algoritmo termina;
 $H \leftarrow H+1$, $R \leftarrow R+1$, $C(R) \leftarrow B(Q)$, $Q \leftarrow Q+1$, $N \leftarrow N-1$ e repetir 06;

5.7.2 - Algoritmo para a operação E.

Algoritmo E. Faz a intersecção de duas listas.

- E1. [Inicialização] $P \leftarrow Q \leftarrow 1$, $R \leftarrow H \leftarrow 0$
 $M \leftarrow$ tamanho de A, $N \leftarrow$ tamanho de B
- E2. Se $A(P) < B(Q)$ então ir para E3 ;
Se $A(P) > B(Q)$ então ir para E4 ;
Se $A(P) = B(Q)$ então $H \leftarrow H+1$, $R \leftarrow R+1$, $C(R) \leftarrow A(P)$, $P \leftarrow P+1$,
 $Q \leftarrow Q+1$, $M \leftarrow M-1$, $N \leftarrow N-1$,
Se $M = 0$ ou $N = 0$, o algoritmo termina. Senão, repetir o passo E2.
- E3. [Avança lista A] $P \leftarrow P+1$, $M \leftarrow M-1$
Se $M = 0$, o algoritmo termina. Senão, ir para E2.
- E4. [Avança lista B] $Q \leftarrow Q+1$, $N \leftarrow N-1$
Se $N = 0$, o algoritmo termina. Senão, ir para E2.

5.7.3 - Algoritmo para a operação MAS NAO.

- M1. [Inicialização] $P \leftarrow Q \leftarrow 1$ $R \leftarrow H \leftarrow 0$ $M \leftarrow$ tamanho de A
 $N \leftarrow$ tamanho de B ;
- M2. Se $A(P) > B(Q)$ então ir para M3 ;
Se $A(P) < B(Q)$ então ir para M4 ;
Se $A(P) = B(Q)$ então $P \leftarrow P+1$, $Q \leftarrow Q+1$, $M \leftarrow M-1$, $N \leftarrow N-1$,
se $N = 0$ então ir para M5, senão, se $M = 0$, o algoritmo termina;

- M3. [Avança Lista B] $Q \leftarrow Q+1$, $N \leftarrow N-1$, Se $N = 0$ então ir para M5, se
não, ir para M2.
- M4. [Elemento de A] $H \leftarrow H+1$, $R \leftarrow R+1$, $C(R) \leftarrow A(P)$, $P \leftarrow P+1$, $M \leftarrow M-1$
Se $M = 0$ o algoritmo termina; senão, ir para M2.
- M5. [Copia Resto de A] Se $M = 0$ o algoritmo termina;
 $H \leftarrow H+1$, $R \leftarrow R+1$, $C(R) \leftarrow A(P)$, $P \leftarrow P+1$, $M \leftarrow M-1$, e repetir M5.

CAPÍTULO VI

INTERPRETAÇÃO E EXECUÇÃO DOS COMANDOS DE BUSCA

6.1 - INTRODUÇÃO.

Este capítulo é dedicado ao reconhecimento da sintaxe e da semântica da linguagem de busca, expressa através de comandos de busca.

A sintaxe é reconhecida diretamente por um analisador sintático, o qual se utiliza de reconhecedores de classes. A medida que vão sendo reconhecidas, as operações são entregues a um sub-sistema de execução, que, na ordem correta, realiza as operações desejadas pelo usuário.

Finalmente, as bibliografias recuperadas são impressas.

6.2 - A ANÁLISE SINTÁTICA.

Devido a simplicidade da linguagem, que possui apenas expressões de busca, nas quais intervêm poucas classes de componentes, ou seja, termos auxiliares, termos-chave (palavras-chave e nomes de autores) e operadores, além de parênteses, a análise sintática é correspondentemente simples, prescindindo de um analisador léxico, ou seja, os componentes básicos são reconhecidos diretamente dentro do contexto, por reconhecedores específicos.

Chamaremos, neste capítulo, de termo, a qualquer componente que tenha associado a si mesmo um conjunto de referências. Enquadram-se nesta designação, portanto, termos auxiliares de busca, palavras-chave e nomes de autor. E por operadores entenderemos qualquer das representações das operações OU, E e MAS NAO.

As figuras 2 e 3 mostram duas representações do analisador sintático, respectivamente um automata finito capaz de reconhecer expressões de busca e um diagrama de fluxo do programa para o analisador. Tanto numa como noutra representação fazemos menção a termos e operadores, que devem ser entendidas dentro das definições acima.

As palavras termo e operador, nas figuras 2 e 3, significam chamadas aos reconhecedores respectivos, descritos com mais detalhes nas seções 6.5.1 e 6.5.2 (termos) e 6.5.3 (operadores).

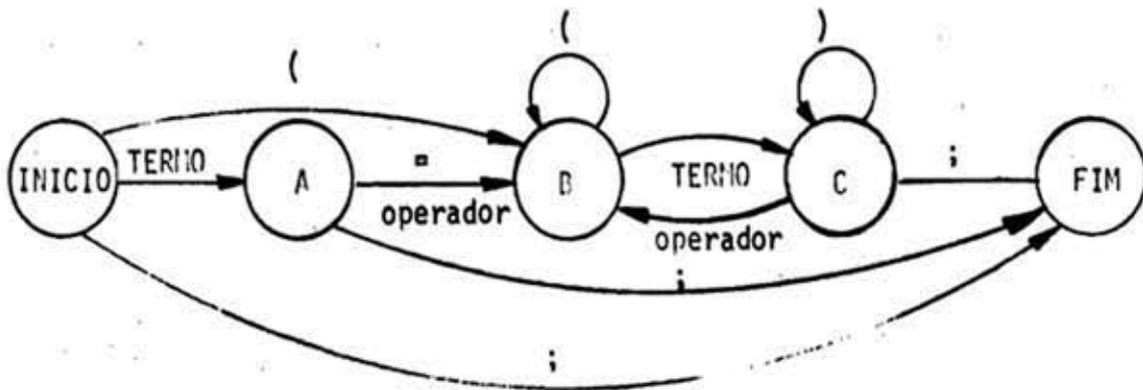


Figura 2

Subentendem-se, na figura 2, os processamentos associados ao reconhecimento dos termos e dos operadores, descrito na seção 6.6. O processamento de termos é realizado nos estados A e C, do automata, e o processamento dos operadores no estado B. Note-se que o limitado contexto onde tem sentido o sinal = como operador de transferencia exigiu a criação do estado A.

O sinal ; indica o fim da expressão. Quando seu aparecimento conduz ao estado FIM, a expressão está correta sintaticamente, e os processamentos realizados a cada mudança de estado calcularam o valor da expressão. No entanto, para cada classe reconhecida na expressão, deve haver

DIAGRAMA DE FLUXO DO ANALISADOR SINTÁTICO

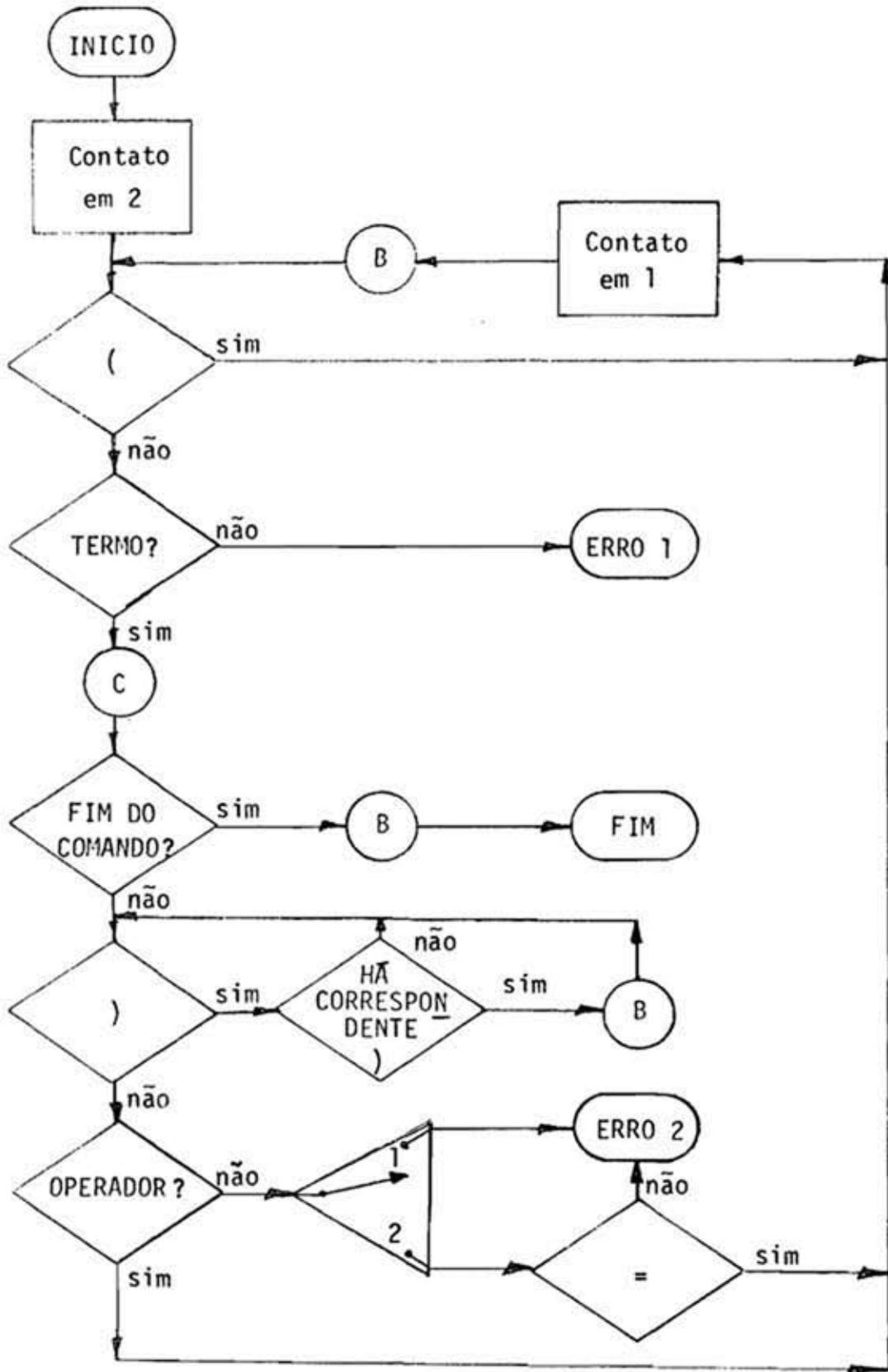


Figura 3

uma mudança de estado. Se isso não for possível, a expressão está incorreta e não pode ser avaliada.

No diagrama de fluxo, os processamentos intermediários são indicados pelos pontos B e C, equivalentes aos estados B e C do automata. Estes pontos ocorrem várias vezes porque são, na verdade, chamadas a rotinas executadas independentemente.

A possibilidade de reconhecimento do sinal = como operador, no diagrama de fluxo, é indicada pela posição da chave, inicialmente no contacto 2, e que passa para o contacto 1 ao ser detetado qualquer operador, inclusive o próprio =.

6.3 - ASPECTO INTERNO DE UM OPERANDO.

Já foi dito que um termo qualquer tem associada a ele uma lista de referências. Por poderem tais listas atingir comprimentos não previsíveis, elas nunca aparecem integralmente na memória do computador durante a execução do programa de busca, ficando sempre em disco. A forma dos arquivos em disco, e seu conteúdo, foram discutidos na Seção 5.5.

Existe, ainda, um quinto arquivo, em disco, com o mesmo formato dos arquivos de "overflow". Neste arquivo armazenamos as listas não vazias que resultam de operações entre outras listas.

De todas as formas, precisamos saber os dados de armazenamento de cada lista, e, estes sim, são guardados na memória, porque são poucos e se modificam com relativa rapidez.

Assim, associamos a cada termo reconhecido pelo analisador os seguintes dados:

a) nome - apenas para termos auxiliares; é o identificador do termo.

- b) tamanho - número de referências contidas na lista.
- c) chave - chave para o acesso aleatório ao primeiro registro das referências; o número de registros é calculado a partir do dado anterior, tamanho.
- d) arquivo - valor convencional, que indica a origem.
 - 1 - palavra-chave
 - 2 - nome de autor
 - 3 - lista resultante de operação
- e) inicialização - valor convencional, que dá indicação sobre o tipo e estado-atual do termo.
 - 0 - não tem lista associada (não inicializado)
 - 1 - tem lista associada (já inicializado)
 - 2 - palavra-chave ou nome de autor
- f) índice - apenas para termos auxiliares; é o índice da entrada correspondente na tabela de termos.

O item e) distingue os termos auxiliares dos termos chave. Ao serem reconhecidos pelo analisador sintático, os termos são ditos não inicializados, pois ainda não se conhece a lista correspondente. Quando esta é determinada, o termo passa ao estado já inicializado, com a consequente atualização desse campo.

6.4 - A TABELA DE TERMOS.

Todos os termos auxiliares definidos num programa de busca são armazenados numa tabela, chamada tabela de termos, ou tabela de símbolos.

Cada entrada nesta tabela contém as informações descritas

na seção anterior, a exceção do item f, cuja função é justamente facilitar o acesso a tabela de termos.

Os termos são colocados sequencialmente na tabela, em ordem cronológica de sua definição. Consequentemente, não guardam nenhuma relação com o índice que lhe coube. Buscas sobre esta tabela são necessariamente sequenciais.

A capacidade da tabela é de cem termos auxiliares, julgada adequada para as necessidades de programas de buscas normais. Como a utilização da tabela deve andar, normalmente por volta de dez a vinte termos, decidiu-se pela organização sequencial da tabela, contra, por exemplo, uma forma de organização aleatória, que empregasse "hash code" na alocação de espaço, o que minimizou o esforço de programação. Para diminuir o número de buscas, cada termo armazenado guarda também o seu número de ordem, de modo que atualizações são imediatas. A técnica adotada exige uma pesquisa sequencial na tabela apenas uma vez por termo por comando.

Termos-chaves, ou seja, palavras-chaves e nomes de autor, não são armazenados. Todas as suas características são calculadas no momento do reconhecimento, e imediatamente utilizadas. Praticamente, são tratados como literais.

6.5 - OS RECONHECEDORES.

6.5.1 - Reconhecedor de termo auxiliar.

É uma rotina extremamente simples, que, a partir de uma posição do comando, conta os seguintes dígitos e letras, levantando assim, o identificador do termo.

Em seguida, o reconhecedor procura na tabela de termos o termo achado. Se já existir, marca-o como já inicializado; se não existir,

coloca-o na tabela, marcando-o como não inicializado. Ainda no segundo caso, estabelece zero como tamanho do termo; para este termo, chave e arquivo não têm sentido. Se o termo já existia, todos os seus dados já tinham sido calculados corretamente.

Os dados do termo reconhecido são colocados juntamente com o índice para a tabela de termos, num local de armazenamento temporário, chamado, por razões que logo se explicam, "elemento-topo-da-pilha".

6.5.2 - Reconhecedor de termo-chave.

Este reconhecedor é mais complicado que o anterior. Inicialmente, tem de identificar a origem do termo: será palavra-chave ou nome de autor, conforme o carácter do comando em observação seja " ou <.

Em seguida extrai todos os caracteres do termo, e investiga o último para saber se se trata de prefixo. Em caso de nome de autor, a condição de prefixo sempre se aplica, explícita ou implicitamente, o que envolve esforço de preparação adicional.

Separados os caracteres do termo, o reconhecedor, que já determinou o arquivo (1 ou 2) e o estado (2) do elemento, determina o tamanho e a chave do elemento. Para isso, realiza uma pesquisa binária sobre o arquivo de descritores ou de autores. Mais uma vez, a condição de prefixo impõe sacrifícios adicionais. É necessário realizar uma múltipla operação OU sobre os elementos abrangidos pelo prefixo, o que implica em comunicação com a rotina de operações, descrita mais adiante. Calculados os OU, a nova lista é gravada no arquivo 3, chave e tamanhos são estabelecidos e o reconhecimento termina.

Se o termo buscado não existe, o tamanho é zero.

Como o outro reconhecedor, este deixa seus dados em elemento-topo-da-pilha.

6.5.3 - Reconhecedor de operadores.

Esta rotina investiga a ocorrência dos caracteres =, /, &, - ou das palavras E, OU e MAS NAO.

= só é considerado operador se for o primeiro operador encontrado, como já foi discutido.

A versão por extenso do operador - só é reconhecida se ocorrerem as duas palavras, MAS e NAO, e nesta ordem.

6.6 - AVALIAÇÃO DA EXPRESSÃO DE BUSCA.

Usamos, para a avaliação das expressões de busca, o método descrito por Samuelson e Bauer, chamado de "Sequential Formula Translation" (referência 15).

Neste processo, os componentes da linguagem, a medida que vão sendo reconhecidos, são entregues a um algoritmo que os processa sequencialmente, ou seja, do mesmo modo em que aparecem na expressão, e em um único passo.

Este algoritmo utiliza duas pilhas (stacks, push-down lists - ver referência 16, capítulo 2), uma para os operandos, no nosso caso os termos, chamada pilha H e outra para os operadores, chamada pilha Z. Para cada operando temos, na pilha, dados relativos a nome, tamanho, chave, arquivo, estado e índice, ou seja, os mesmos dados que os reconhecedores de termos colocaram no campo "elemento-topo-da-pilha", que foi assim chamado porque os dados passam por ele antes de atingir o topo da pilha. A pilha dos operadores contém um único caracter por entrada. Além dos operadores = / & e -, podemos ter também, nesta pilha, parênteses esquerdo.

Na figura 3, os blocos B e C correspondem, respectivamente,

ao processamento de operadores e de termos. O processamento de termos \bar{e} , sistematicamente, a colocação dos dados do termo na pilha H. Já o processamento dos operadores determina a ordem pela qual as operações devem ser efetuadas para produzir o resultado procurado. Esse processamento \bar{e} indicado esquematicamente na figura 4.

A operação do processador de termos, como dissemos, \bar{e} extremamente simples, sendo descrita basicamente como segue:

$$\begin{aligned}h &\leftarrow h+1 \\ H_h &\leftarrow \text{elemento-topo-da-pilha}\end{aligned}$$

onde h \bar{e} o índice para a pilha H, em geral indicando o seu topo, e H_h \bar{e} o conteúdo da célula de ordem h da pilha H.

O processador de operadores \bar{e} mais complicado, pois \bar{e} ele quem determina a ordem de execução das operações; para isso, alguns testes são necessários, os quais estão mostrados, esquematicamente, na figura 4.

A notação empregada neste fluxo \bar{e} a seguinte:

s - índice para a pilha Z. Indica o topo da pilha.

Z_s - conteúdo da célula de ordem s da pilha Z.

α - operador recém-reconhecido. Pode ser = & / - () ou ;. Este último \bar{e} signa o fim do comando.

A figura 4 \bar{e} uma representação fiel do algoritmo, tal como foi implementado no computador. O entendimento do algoritmo fica facilitado se o explicarmos através de uma tabela, onde indicamos a ação tomada frente aos diversos pares α, Z_s , ou seja, o operador recém-reconhecido e o operador no topo da pilha Z.

PROCESSAMENTO DE OPERADORES

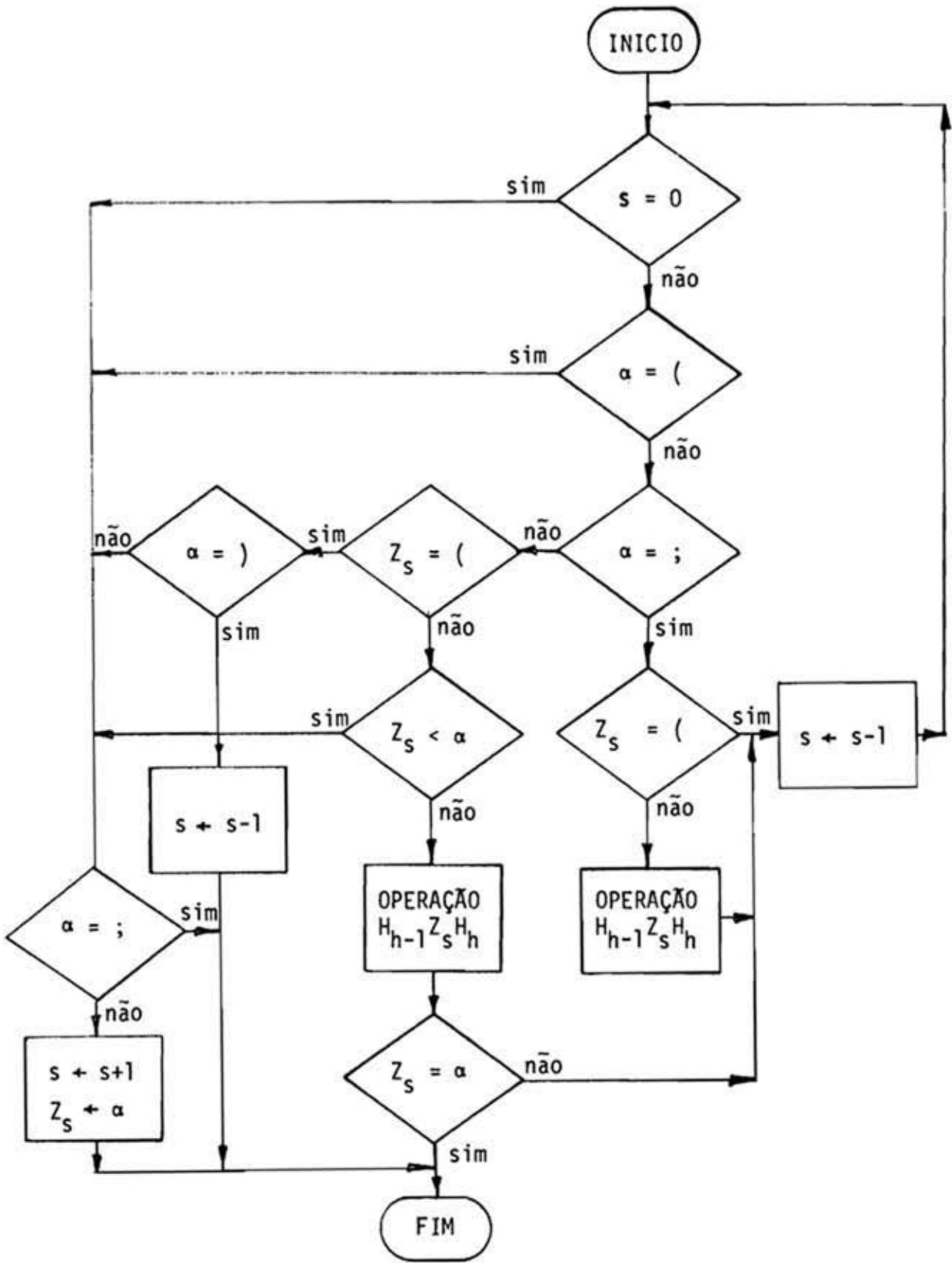


Figura 4

Z_s	α	
ϕ	qualquer	O operador α é colocado no topo da pilha. Entretanto, se α indicar o fim do comando, a rotina termina. Ex.: A & ...
/	/	O operador em α é da mesma precedência que o operador em Z_s . Na linguagem de busca, como não há dois operadores diferentes com a mesma precedência, α é igual a Z_s . Ex.: $\begin{array}{ccc} & Z_s & \alpha \\ & \downarrow & \downarrow \\ A & \& B & \& \dots \\ A & - & B & - & \dots \\ A & / & B & / & \dots \end{array}$
&	&	
-	-	
		A operação $H_{h-1}Z_sH_h$ é realizada, o resultado ficando em H_{h-1} ; conseqüentemente, há um rearranjo na pilha H; a pilha Z não se altera.
(& - / (O operador em α é de maior precedência que o operador em Z_s , ou, pela presença de "(" nas imediações, a operação indicada em α deve ser realizada antes daquela indicada por Z_s . Ex.: $\begin{array}{ccc} & Z_s & \alpha \\ & \downarrow & \downarrow \\ A & \& (& B & / & \dots \\ A & = & B & - & \dots \\ A & / & B & \& \dots \\ A & \& B & - & \dots \\ A & = & (& \dots \end{array}$
=	/ & -	
/	& -	
&	-	
= / & -	(
		α é colocado no topo da pilha Z. Naturalmente, a capacidade da pilha Z é verificada (admite-se um máximo de 20 operadores na pilha).

Z_s	α																															
() ;	<p>O elemento topo da pilha Z é eliminado, com consequente reajuste do índice S.</p> <p>Se $\alpha = ";"$, todo o processo é repetido. Lembramos que se convencionou que o fim do comando fecha todos os parênteses ainda abertos. Entretanto, se $\alpha = ")"$, a rotina termina.</p>																														
-	/ &	<p>O operador em Z_s é de precedência maior que o operador em α, ou este indica de alguma forma que a operação Z_s deve ser realizada.</p> <p>Ex.:</p> <div style="text-align: center;"> <table style="border: none;"> <tr> <td></td> <td style="text-align: center;">Z_s</td> <td></td> <td style="text-align: center;">α</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">↓</td> <td></td> <td style="text-align: center;">↓</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">A</td> <td style="text-align: center;">-</td> <td style="text-align: center;">B</td> <td style="text-align: center;">&</td> </tr> <tr> <td></td> <td style="text-align: center;">A</td> <td style="text-align: center;">&</td> <td style="text-align: center;">B</td> <td style="text-align: center;">/</td> </tr> <tr> <td></td> <td style="text-align: center;">A</td> <td style="text-align: center;">-</td> <td style="text-align: center;">B</td> <td style="text-align: center;">)</td> </tr> <tr> <td></td> <td style="text-align: center;">A</td> <td style="text-align: center;">=</td> <td style="text-align: center;">B</td> <td style="text-align: center;">;</td> </tr> </table> </div> <p>A operação $H_{h-1}Z_sH_h$ é realizada; o resultado fica em H_{h-1}, com consequente reajuste da pilha H. O operador Z_s é eliminado, com consequente reajuste da pilha Z, e todo o processo é repetido.</p>		Z_s		α			↓		↓			A	-	B	&		A	&	B	/		A	-	B)		A	=	B	;
	Z_s			α																												
	↓			↓																												
	A		-	B	&																											
	A	&	B	/																												
	A	-	B)																												
	A	=	B	;																												
&	/																															
/ & -)																															
= / & -	;																															

6.7 - A MATERIALIZAÇÃO DAS OPERAÇÕES.

Durante o processamento dos operadores, surge a oportunidade para combinar duas listas numa terceira, através da realização de uma operação.

Esta tarefa é realizada por uma rotina em separado das demais, chamada pelo processador de operadores. Não há necessidade de transferência de parâmetros, visto que os elementos a operar estão no topo da

pilha H e a operação propriamente dita está representada no topo da pilha Z.

A rotina de operação encontra, portanto, com facilidade, os dados a manipular. Os dois elementos do topo da pilha H são eliminados, e o resultado, uma vez encontrado, vai para o topo da mesma pilha. A rotina, portanto, produz a correta acomodação da pilha H, eliminando e criando elementos. Não mexe, porém, na pilha Z, que é da alçada exclusiva do processador de operadores.

Além das operações E, OU e MAS NAO, esta rotina realiza ainda a operação =, que consiste basicamente em transferir os dados do operando B para os do operando A, eliminando aqueles da pilha e deixando estes no topo.

Lembramos, mais uma vez, que na pilha H estão apenas dados genéricos dos termos, as referências propriamente ditas residindo em disco, com endereço conhecido.

Alguns erros são detetados nesta fase do programa, e são aqueles que dizem respeito ao estado de inicialização de termos auxiliares empregados nas expressões. Estes erros são de dois tipos: uso, como operando, de um termo auxiliar não inicializado, e definição de termo já inicializado. Nos dois casos, a rotina toma decisões, arbitrárias, a respeito do assunto, e prossegue de forma a não truncar a pesquisa bibliográfica. No primeiro erro, associamos um tamanho zero à lista, que na verdade não existe. No segundo, a definição incorreta é abandonada. Em qualquer dos casos, o usuário é informado por meio de uma mensagem.

Os algoritmos para a construção das listas resultado foram apresentados no capítulo V, tendo sido implementados exatamente como foram descritos, para listas não vazias.

Quando a operação envolve listas vazias, adotam-se os resultados apresentados a seguir:

$$\begin{aligned} A / \phi &= A \\ \phi / B &= B \\ A \& \phi &= \phi \& B = \phi \\ A - \phi &= A \\ \phi - B &= \phi \end{aligned}$$

Para estes casos, o resultado é imediato. Para os demais, é realmente necessária a execução dos algoritmos.

6.8 - OBTENÇÃO DAS LISTAS OPERANDO E CRIAÇÃO DA LISTA RESULTADO.

Como já foi dito anteriormente, a lista de referências de um termo chave só eventualmente fica toda na memória. Isso sucede quando a lista é pequena. Da mesma forma, a lista resultante de uma operação também não fica integralmente na memória; de qualquer maneira, de nenhum modo temos todas as listas na memória. Queremos dizer que é parte do processo de recuperação dessas listas a sua localização no disco e subsequente leitura.

Além disso, devemos notar que a lista de um operando pode provir de diferentes arquivos: de descritores, de autores, de resultados; além disso, os dois iniciais continuam em dois outros arquivos diferentes, os de overflow, o que leva a cinco o número de fontes diferentes de números de referências. Ademais, os registros não são de mesmo tamanho, e portanto, o número de referências encontradas com uma leitura física varia, dependendo do arquivo.

A inclusão desse mecanismo de obtenção de referências nos algoritmos das operações complicaria por demais estes últimos. Decidimos, então, escrever rotinas a parte, que fornecem às operações, a pedido, as referências, uma a uma, e que sinalizam o término da lista.

Raciocionando analogamente, fizemos os algoritmos entregarem a uma terceira rotina, um a um, os componentes da lista resultado. Es

ta rotina, então, compõe a lista final, e, em momentos próprios, grava pe da ços dessa lista em disco, conforme o tamanho adotado para o registro.

6.9 - TEMPOS DE EXECUÇÃO.

O fator dominante nos cálculos da duração de um programa de busca é o disco, meio mecânico, e, portanto, bastante lento em compara ção com o processador central. Como o disco é um dispositivo aleatório, qualquer estimativa de tempo deve girar sempre em torno de valores médios.

A operação elementar mais demorada, considerando o volume de informação presente nos testes, é o reconhecimento de uma palavra-cha ve, e o conseqüente estabelecimento dos dados a seu respeito.

De fato, considerando a existência de 17.000 descritores e de 10.000 autores distintos, o processo de pesquisa binária adotado nece sita rã de 14 acessos para o primeiro arquivo e de 13 acessos para o segun do. O tempo médio por acesso é de 20 mili-segundos, o que nos dá 280 e 260 mseg de duração média total para cada tipo de termo chave. Consideran do um sempre presente tempo de processamento, temos um tempo médio de 300 mseg por termo-chave.

A realização de uma operação produz uma lista, que deve ser gravada em disco, o que consumirá mais 20 mseg, em média. A esse tempo so ma se o tempo de obtenção das listas dos operandos, 20 mseg para cada um, totalizando em média 60 mseg. Considerando o tempo de processamento inter no, e o fato de que algumas listas ocupam mais de um registro, exigindo, portanto, mais de um acesso, podemos, pessimistamente, determinar 100 mseg como sendo a duração média de uma operação.

Não há nenhuma indicação de quantas operações e termos-cha ves comporão o comando de busca típico. Admitindo duas operações e dois termos-chave com um valor provável, chegamos a 800 mseg como a duração esperada de um comando de busca.

Este tempo é muito bom até mesmo para um sistema "on-line", no qual o computador interage com o usuário. Teríamos um tempo de espera menor que um segundo, desprezível em escala humana.

Para um sistema orientado para o processamento em "batch", como é o nosso, estas estimativas garantem um processamento rápido, no qual os recursos do computador ficarão presos por pouco tempo. A certeza de que o processo de recuperação tomará pouco tempo, mesmo para pesquisas inutilizadas por erros de sintaxe, nos levou a seguir a linha mais simples de programação, como comentamos a seguir.

Durante a análise dos comandos, todos os cuidados foram tomados para impedir a terminação abrupta do processo de interpretação; em vários casos, hipóteses sobre as intenções do usuário foram feitas, permitindo que se chegasse a um resultado, senão o procurado, pelo menos o mais próximo possível dele.

Por vezes, entretanto, não é cabível o emprego de nenhuma técnica de recobrimento de erro, e o comando é abandonado. Se se tratasse de uma definição de termo, futuras referências a esse termo estarão em erro, pois não foi possível inicializá-lo. Em consequência, o programa receberá inúmeras mensagens de erro.

Além das mensagens, a ocorrência de um erro consumirá tempo em pesquisas binárias e operações, de certo modo inúteis.

O tempo assim perdido poderia ser economizado se modificássemos o método de análise, separando a análise sintática da execução do comando, e somente executando a segunda parte se a primeira revelasse um comando isento de erros.

Essa separação pode ser feita a nível de comando ou se aplicar a todo o programa de busca.

No primeiro caso o ganho seria pequeno, porque a incapacidade de definir um termo não impediria a tentativa de usá-lo mais abaixo.

O segundo caso complicaria sobremaneira o programa. Inicialmente, a sintaxe da linguagem é tão simples que não valeria a pena traduzi-la para um código intermediário, mais simples; este seria praticamente igual ao código original. Depois, o comprimento de um programa é, em teoria, ilimitado. Para analisar duas vezes os comandos, seria necessário armazenar e recuperar os comandos, o que criaria mais um arquivo em disco. Feitas as contas, a economia seria pequena.

Com base no raciocínio acima, em que entram em conflito ganho de tempo de execução e simplicidade de programação, decidimos dar ganho de causa ao segundo. Ajudou-nos, nesta decisão, a expectativa de que erros serão pouco comuns, dada a simplicidade da sintaxe.

6.10 - IMPRESSÃO DAS REFERÊNCIAS RECUPERADAS.

O verdadeiro resultado que se procura com a aplicação desse sistema é um conjunto de referências, que satisfaçam ao interesse do usuário.

Este produto final é apresentado na forma de um relatório, impresso diretamente pelo computador, contendo o material recuperado, ou seja, as referências. Exemplos são encontrados nos apêndices B e C.

Na verdade, o sistema produz dois relatórios. No primeiro é reproduzido (listado) o programa de busca, com todas as mensagens de erro e/ou avisos relativos a decisões tomadas pelo programa.

O segundo relatório é a bibliografia encontrada para cada uma das buscas estabelecidas no programa. Para evitar a impressão de bibliografias muito extensas, as quais, em geral, não têm grande significação

do para o usuário, existe um limite para o número máximo de referências impressas, o qual nunca é ultrapassado. Quando a bibliografia é muito extensa, das referências recuperadas sorteiam-se algumas, por um processo pseudo-aleatório, as quais são impressas até ser atingido aquele limite.

CAPÍTULO VII

CONCLUSÕES

Mostramos, nos capítulos precedentes, os diversos componentes de um sistema de recuperação de informação, sua estrutura e as interligações de cada um deles com os demais.

Embora desenvolvido especificamente para computadores B-3500 e B-6700, este sistema é razoavelmente universal, por ser constituído por um único programa, codificado integralmente em Cobol, e no qual foram empregadas apenas as construções gerais dessa linguagem de programação. A existência de um compilador Cobol em praticamente todos os computadores modernos, torna o SIRIUS facilmente transportável para outros modelos.

Nos detalhes, porém, nota-se que se trata de uma implementação piloto, a qual, com tempo e mais recursos, pode evoluir em sofisticação.

Para computadores equipados com dispositivos de teleprocessamento, o sistema poderia ser adaptado para admitir consultas remotas, através do uso de teletipos ou de unidades "display", equipadas com tubos de raios catódicos. Ainda nesse caso poderíamos evoluir de um atendimento por vez para o atendimento simultâneo a várias consultas. Naturalmente, o fluxo de processamento se ajustaria, permitindo ao usuário modificar o seu programa de busca dinamicamente.

A quantidade de referências recuperáveis pode se estender "ad infinitum", mediante a incorporação de novas unidades de armazenamento. Modelos novos, com capacidade para várias dezenas de milhões de ca

racteres podem ser adicionados à configuração atual das máquinas. Grosseiramente, uma referência ocupa, em média, 150 caracteres. Uma única unidade de discos, removível, com 60M caracteres de capacidade, armazenaria 400.000 referências bibliográficas, o que corresponde a um volumoso acervo.

Finalmente, a expansão de hardware que complementaria o banco de dados: leitoras de microfilme remota e automaticamente controladas. Após localizar as referências desejadas, o usuário poderia, imediatamente, ler o resumo (abstract) do artigo, ou até mesmo o artigo integral, com quaisquer figuras ou gráficos que possuísse.

Claro que todas as alterações no hardware exigiriam correspondentes modificações no programa. Estas seriam adições mais que modificações, visto que falta, no momento, habilidade ao programa para lidar com estes periféricos. O núcleo do programa, porém, o interpretador da linguagem, os diversos reconhecedores, permaneceria inalterado.

Melhorias no sistema, de caracter conceitual, porém, estão mais ligadas à parte de documentação. Uma preparação mais cuidadosa das referências, e a obtenção das palavras chave de um thesaurus, e não diretamente do título seriam providências cabíveis, que proveriam ao usuário uma melhor indexação.

Pode-se, também, estender a linguagem para recuperar não apenas palavras chave e nomes de autor, mas também outros dados como, por exemplo, ano de publicação, local de publicação, editor, etc. Essa extensão seria muito natural.

Um outro tipo de extensão da linguagem serviria para os casos em que o programa fosse usado para recuperar não referências bibliográficas, mas, digamos partes e material de um almoxarifado. Admitamos que cada peça cadastrada seja descrita por diversos dados, como nome, proce

dência, material, peso, preço, localização nas prateleiras, data de entrada no estoque e outros mais. Alguns desses itens são puramente numéricos, como preço e peso. A recuperação desses itens admite uma relação, como a procura de todos os itens de peso menor que um quilo.

A linguagem de busca poderia ser estendida para reconhecer expressões como

```
MATERIAL = "COBRE" E PESO > 1
```

Finalmente, a última sugestão. Certamente foi notado que, ao dividirmos o cartão em três campos, um deles foi reservado para um número de ordem do cartão. Esta reserva foi feita prevendo a utilização futura de um meio rápido de armazenagem e correção de pesquisas. O usuário numeraria seus cartões e o programa os gravaria em disco; posteriormente, alterações seriam feitas baseadas no campo de numeração: seriam possíveis substituições, inclusões e retiradas de comandos. Desta maneira, estaria sempre presente a pesquisa (ou pesquisas) de um usuário.

Na situação presente, pouco proveito se pode tirar de um tal arranjo, visto que os dados são permanentes, não se modificam. A variação ocorre justamente nas pesquisas, donde ter uma pesquisa pré-fixada pouco adiantaria.

Suponhamos, entretanto, que as referências sejam extremamente voláteis; por exemplo, mensalmente se relacionariam os artigos publicados em determinados periódicos. A pesquisa fixa corresponderia, agora, a um perfil de interesse, o qual continuaria flexível, podendo ser alterado a qualquer momento. Mudado o data-base, far-se-iam as buscas adequadamente preparadas, informando-se os usuários do resultado obtido.

Ou seja, passamos a ter um sistema de disseminação seletiva de informação (SDI).

BIBLIOGRAFIA

1. Fischer, M. - The KWIC index concept: a retrospective view. *American Documentation* 17(2) 57-70, abril 66.
2. Luhn, H.P. - Keyword-in-Context Index for Technical Literature (KWIC Index), RC-127. IBM Corporation, Yorktown Heights, N.Y., agosto de 69. Também, em *American Documentation*, 11(4) 288-95, outubro 60.
3. Estados Unidos. Library of Congress - The MARC pilot experience; an in formal summary. Washington, 1968.
4. Estados Unidos. Library of Congress - The MARC II format. Washington, 1968.
5. National Science Library, National Research Council of Canadá. Profile Design Manual, Ottawa, abril 70.
6. Garfield, E. - Citation Indexing for Studying Science - *Nature*, vol. 227, 669-671, agosto de 1970.
7. Medeiros, H. - A automação a serviço da documentação. Rio de Janeiro, Instituto de Pesquisas Rodoviárias, 1970.
8. Taube, M. et al - Unit terms in coordinate index. *American Documentation* 3(4) 213,8 - outubro 1952.
9. Mooers, C.N. - Descriptors. In - *Encyclopedia of Library and Information Science*. New York, Marcel Dekkar Inc., 1972.
10. Zaher, C., Guimarães Y. e Teixeira, Iberê L.R. - O sistema integrado de automação das bibliografias especializadas brasileiras - Trabalho

apresentado a 11.^a reunião da FID/CLA, Lima, Peru, setembro de 1971.

11. Senado Federal - Prodasen - setembro de 1972.
12. IBBD/CBPF - Bibliografia Brasileira de Física, 1961-67. Rio de Janeiro, 1968.
13. Black, D.V. e Farley, E.A. - Library Automation. In - Annual Review of Information Science and Technology. Carlos Cuadra, Editor. New York, 1966.
14. Chastinet, Y. - Manual de referência bibliográfica para o SIABE - IBBD, Rio de Janeiro, 1972.
15. Samuelson, K. e Bauer, F.L. - Sequential Formula Translation - In - Saul Rosen, editor, Programming Systems and Languages, New York, 1967.
16. Knuth, D.E. - Fundamental Algorithms - Reading, Massachusetts, 1968.
17. Naur, P., editor - Revised report on the Algorithmic Language Algol 60. In - Saul Rosen, editor, Programming Systems and Languages, New York, 1967.
18. Frota, Lia M. de A. e Nunes, R.P. - Emprego de sistema eletrônico na elaboração de catálogos de biblioteca. Trabalho apresentado a 9.^a reunião da FID/CLA, Rio de Janeiro, 1969.
19. Burroughs Corporation - B-3500 Cobol Reference Manual form. 1033099. Detroit, 1971.
20. Burroughs Corporation - B-6700 Cobol Reference Manual form. 5000656. Detroit, 1973.
21. Burroughs Corporation - B-6700 Extended Algol Language Reference Manual form. 5000128. Detroit, 1971.

APÊNDICE A

REPRESENTAÇÃO DOS NOMES DE AUTOR

As entradas relativas aos nomes dos autores são compostas segundo as regras estipuladas na referência 14. Para aplicação no SIRIUS, foi necessário, porém, fazer algumas adaptações, de modo a unificar os nomes de autores, pois a mesma pessoa assina seus trabalhos de diversas maneiras, e, ao mesmo tempo, simplificar as entradas, para diminuir o risco de falsa indexação.

A adaptação consistiu em transformar os nomes de autores presentes no SIABE em entradas recuperáveis compostas por apenas um sobrenome e por número variável de iniciais. Estas últimas, quando existem, são separadas do sobrenome por um único espaço em branco. As iniciais, por sua vez, apresentam-se na ordem natural, uma depois da outra, sem nenhum sinal que as caracterize, entre elas. Foram eliminados, assim, as vírgulas, pontos e espaços preconizados pelo SIABE.

Além dessas adaptações, faz-se também:

- em casos de sobrenomes compostos, é feita uma entrada para cada sobrenome.

Pimentel Gomes, A.J.

Pimentel AJ

Gomes AJ

Magalhães Costa

Magalhães

Costa

- quando uma das palavras do sobrenome composto é uma entre Filho, Junior, Sobrinho, Neto ou Netto, não é feita a entrada correspondente.

Pinto Neto, C.H.
Silva Junior

Pinto CH
Silva

- preposições, conjunções e contrações destas com artigos, são eliminadas.

Anjos, J.A. dos
Silva, A.L. e

Anjos JA
Silva AL

- siglas, congressos ou reuniões, quando citadas como autor, são aproveitadas apenas quando formadas de uma única palavra.

APÊNDICE B

EXEMPLOS DE PESQUISAS COMPLETAS

Este apêndice é composto por reproduções das saídas do computador, em resposta a comandos de busca.

Todas as buscas são apresentadas sem erros. As mensagens do sistema quando da detecção de alguma incorreção são discutidas no apêndice C. No entanto, comentamos algumas das particularidades do sistema nas próprias listagens, na forma de comentários, eventualmente realçadas por indicações visuais.

O programa produz duas listagens por pesquisa: a listagem do programa de busca, com mensagens de erros e avisos, e, se houver alguma bibliografia recuperada, esta é listada.

No primeiro relatório, todos os cartões são listados separando-se os três campos por uma coluna em branco.

As referências encontradas são integralmente listadas, logo abaixo de um cabeçalho-resumo. Além disso, as referências são numeradas sequencialmente.

*** S I R I U S *** LISTAGEM DO PROGRAMA DE BUSCA *** INPE/DPU ***

PESQUISA SEM NOME

USUARIO DESCONHECIDO 30 JUN 73 21,19,21 ENDERECO IGNORADO

A PRESENTE PESQUISA FOI INDIVIDUALIZADA COM UM CARTAO S INTEIRA-
MENTE EM BRANCO, COMO CONSEQUENCIA, O CABECALHO CONTEM MENSAGENS QUE
INDICAM A INTENCAO DO USUARIO EM PERMANECER ANONIMO.

ESSE EFEITO, EVIDENTEMENTE, INTENCIONAL, ANTE O PESSIMO RESUL-
TAO, ESPERAMOS QUE OS USUARIOS CODIFIQUEM CORRETAMENTE ESTE CARTAO,
PERMITINDO UMA DEVOLUCAO SEM PROBLEMAS,

"MEDICINA"

CONSULTA ASSUSTADORAMENTE AMPLA, A QUAL TEM UMA RESPOSTA MUITO
GRANDE, APESAR DA LIMITAÇÃO, HA ELEMENTOS NA MOSTRA CAPAZES DE AJUDAR
O USUARIO A EXPOR MELHOR SEU INTERESSE,

*** S I R I U S *** LISTAGEM DAS REFERENCIAS ENCONTRADAS *** INPE/OPD ***

PESQUISA SEM NOME

USUARIO DESCONHECIDO 30/JUN/73 21,19,21 ENDEREÇO IGNORADO

4113 REFERÊNCIA(S) ENCONTRADA(S) PARA A PESQUISA "MEDICINA"
DESSAS, APENAS 15 SERÃO LISTADAS,

- 1 KULISHI, M.
A IUNTOFORESE COMO FATOR TERAPEUTICO
ARS CURAN, 3(5)76-80, JUL, 1970
- 2 LAGES NETTO
DISTURBIO RESPIRATORIO DO RECEM-NASCIDO
F, MED, 60(1)65-71, JAN, 1970
- 3 MARINHO, S.D. TAVARES, B.M.
MANITOL, MECANISMO DE AÇÃO, AVALIAÇÃO EXPERIMENTAL DAS ALTERAÇÕES NO EPITELIO
TUBULAR RENAL
O HOSPITAL 77(2)473-84, FEV, 1970
- 4 TEIXEIRA, L.C.V. LIMA, V.B. DE A. ZANON, U.
TUBERCULOSE GANGLIONAR PROVOCADA POR MICOBACTERIA DO GRUPO AVIUM-BATTEY
J, BRAS, MED, 18(3)61-2, MAR, 1970
R, BRAS, MED, 27(11)537-8, NOV, 1970
- 5 CANVALHOSA, A.S.B. DE
ALTIMETRO PARA CÁLCULO DIRETO DA ALTURA DA PROTESE A SER USADA NA CIRURGIA DO
ESTRIBO
ARS CURAN, 3(8)18-20, OUT, 1970
- 6 CORREA, A. AMATO NETO, V. SERRANO, C. DA P.
PESQUISA DE CASOS DE PARA COQUELUCHE ENTRE CRIANÇAS RESIDENTES EM HABITAÇÃO
COLETIVA DA CIDADE DE SÃO PAULO
R, MED, IAMSPE 1(1)6-8, JAN./MAR, 1970
- 7 SATHLEH, R.
BASES FISIOLÓGICAS E FARMACOLÓGICAS DA CORTICÓIDEA TERAPIA
R, FARM, E ODONT, 35(331)35-45, SET, 1968
- 8 KAMEL, D.
A CIENCIA DE EMAGRECER
RIO DE JANEIRO, CULTURA MEDICA, 1970, 144 P.
- 9 SAN JUAN, F. TREIGER, M. GONCALVES, A.J.R.
CONTRIBUIÇÃO AO ESTUDO DA ICTERICIA DA LEPTOSPIROSE ICTEREMORRAGICA
B, CENTRO EST. MOSP, SERV, EST, 22(4)253-64, OUT./DEZ, 1970

- 10 SAAD, M.A, COVIAN, M.R.
 INTERACAO ENTRE HIPOTALAMO E AMIGDALA NO CONTROLE DA INGESTAO DE SODIO NO RATO
 IN - REUNIAO ANUAL DA SBPC, XXII, SALVADOR, 1970
 RESUMOS
 SAO PAULO, SOCIEDADE BRASILEIRA PARA O PROGRESSO DA CIENCIA, 1970, P. 376-7,
 RESUMO
- 11 KATAYAMA, M, FREITAS, J.A.M, FERREIRA, A.A,
 HALOTHANE E PRESSAO INTRA OCULAR
 M, BRAS. OFTALMOL, 29(4)373-8, DEZ, 1970
- 12 SANTOS-SILVA, M.A, DOS
 A MINHA ESTANHA HIDRONEFROSE (1944)
 IN - SANTOS-SILVA, M.A, DOS
 TITULOS E TRABALHOS APRESENTADOS A FUNDAÇÃO ESCOLA DE MEDICINA E CIRURGIA DO RIO
 DE JANEIRO
 RIO DE JANEIRO, 1969, P. 166-72, RESUMO
- 13 LEITE, D.C.
 O MEDICO PLANTONISTA SUAS ATRIBUICOES
 R, PAUL. HOSP, 18(5)41-2, MAIO 1970
- 14 NUNES, D.M, KARLING, A,
 FUNCAO DA ENFERMAGEM NO CONTROLE DAS INFECCOES, INFECCAO HOSPITALAR EXPERIENCIA
 DA COMISSAO DE INFECCAO DO HOSPITAL ERNESTO DORNELLES
 IN - SEMANA DE ESTUDOS DO HOSPITAL ERNESTO DORNELLES, 2, PORTO ALEGRE, 1969
 ANTIBIOTICOS E QUIMIOTERAPIA ANTIBACTERIANA
 PORTO ALEGRE, CENTRO DE CULTURA HOSPITAL ERNESTO DORNELLES, 1970, P. 295-315
- 15 LIMA, D.G, DE D'ALBUQUERQUE, I.L, MACIEL, G.M,
 SUBSTANCIA ANTI MICROBIANAS DE PLANTAS SUPERIORES, XXIX, PRIMEIRAS OBSERVACOES
 SOBRE A ATIVIDADE ANTI MICROBIANA DE CELASTROL
 R, INST. ANTIB, 9(1/2)75-7, DEZ, 1969

```

*****
*                                     *
*   *** S I R I U S ***   LISTAGEM DO PROGRAMA DE BUSCA   *** INPE/DPD ***   *
*                                     *
*                                     *
*                                     *
*                                     *
*   SEBASTIAO PEREIRA 30/JUN/73   21,19,21   ENDEREÇO IGNORADO   *
*                                     *
*****

```

DESTA VEZ JÁ CONHECEMOS O NOME DO USUÁRIO, LOGO TEREMOS INDICAÇÕES MAIS COMPLETAS.

@ <COELHO> "PELE"

EM UM ÚNICO CARTÃO FORAM FORMULADAS DUAS BUSCAS, UMA SOBRE AUTORES E A OUTRA SOBRE PALAVRAS CHAVE.

PESQUISA SEM NOME

SEBASTIAO PEHEIRA 30/JUN/73 21,19,21 ENDEREÇO IGNORADO

AS REFERÊNCIA(S) ENCONTRADA(S) PARA A PESQUISA <COELHO>
SÓSSAS, APENAS 15 SEÃO LISTADAS.

- 1 COELHO, L. ASSIS, L.M.
ESTUDO CLÍNICO DO TRABALHO MENTAL DE 62 PACIENTES EPILEPTICOS ADOLESCENTES
ARQ. NEUROPSIQ. 28(2)135-40, JUN, 1970
- 2 MEHSZENHUT, E.P. FRANCO, L.G.P. COELHO, M.S.
SOBRE UM CASO DE HEMO GLOMERULOPATIA PAROXISTICA NOTURNA E GRAVIDEZ
J. BRAS. GINECOL. 89(2)89-100, FEV, 1970
- 3 BONOMO, I. BATISTA, R. COELHO, S.R. MARINHO, M.A.A.
LABORATÓRIO EM DOENÇAS REUMÁTICAS
J. BRAS. MED. 18(2)15-36, FEV, 1970
- 4 BUNHEM, M.S. DE A. LAUN, I. COELHO, R. SOUSA, P.L.R.
HIPER TIREOIDISMO NA INFANCIA, REVISÃO A PROPOSITO DE UM CASO COM 16 MESES
ARQ. BRAS. ENDOCRINOL. E METABOL. 18(2)107-20, AGO, 1969
- 5 COELHO, A.C. DOS S. MATHEWS, M.M.
SUPERFAMÍLIA TONNACEA, I, FAMÍLIA BURSIIDAE BURSA-(COLUBRELLINA) NATALENSIS SP. N.
(MOLLUSCA, GASTROPODA)
B. MUS. NAC. NOVA SEB., ZOOLOGIA (279)1-6, DEZ, 1970
- 6 COELHO, D.T.
HIGROMETRIA
SEIVA 30(69)53-76, JUN, 1970
- 7 CHAVES, G.M. CHUZ FILHO, J. DA CARVALHO, M.G. DE MATSUOKA, K. COELHO, D.T.
SHIMIZU, C.
A FERRUGEM DO CAFEEIRO (HEMILEIA-VASTATHIX-BENK.-HM.) REVISÃO DE LITERATURA COM
RESENHAS E COMENTÁRIOS SOBRE A ENFERMIDADE NO BRASIL
SEIVA 30(1)-75, DEZ, 1970
- 8 COELHO, A.S.M. MELLO, H.A. SIMÕES, J.W.
COMPORTAMENTO DE ESPÉCIES DE EUCALIPTOS FACE AO ESPACAMENTO
IPEF (1)29-45, 1970
- 9 COELHO, M.
LAMA ASFÁLTICA, ECONOMIA NA CONSERVAÇÃO DE PAVIMENTOS
DIREC. CONSTR. 6(11)33-40, AGO, 1970

- 10 COELHO, P.A.
LISTA DOS STENOPOURIA (CHUSTACEA DECAPODA NATANTIA) DE PERNAMBUCO E ESTADOS
VIZINHOS
IMB, OCEANOGR, UNIV. FED. PERNAMBUCO (9/1)249-54, 1967/69
- 11 SELIANE, H.P., CAMPOS, M.G. DE GUINARRES, M., SIQUETRA, C.S., ALMEIDA, P.S. DE
COELHO, R.R., TUNELLO, M.R. DE G.
OBTENCAO DE OXIDO DE BERILIO
IN - REUNIAO ANUAL DA SHPC, XXII., SALVADOR, 1970
RESUMOS
SAO PAULO, SOCIEDADE BRASILEIRA PARA O PROGRESSO DA CIENCIA, 1970, P. 184.
RESUMO
- 12 SILVA, A.G. DA COSTA, A.M., COELHO, M.P.
CONTRIBUICAO AO ESTUDO DA ENCEFALU MIFLITE AVIARIA NO ESTADO DA BAHIA
B. INST. BIOL. BAHIA 9(1)26-8, 1970
- 13 LIMA, O.G. DE LYRA, F.D. DE A., ALBUQUERQUE, M.M.F. DE MACIEL, G.M.,
COELHO, J.S. DE H.
PRIMEIRAS OBSERVAÇÕES SOBRE O COMPLEXO ANTIBIOTICO E ANTI TUMORAL, META-ICINA,
PRODUZIDO PELO STREPTOMYCES-OLINDENSIS NOV. SP. (AUFPE INSTITUTO DE ANTIBIOTICOS
DA UNIVERSIDADE FEDERAL DE PERNAMBUCO
B. INST. ANTIB. 9(1/2)27-37, DEZ. 1969
- 14 CHIARINI, J.V., COELHO, A.G. DE S.
COBERTURA VEGETAL E NATURAL E AREAS REFLORESTADAS DO ESTADO DE SAO PAULO
B. INST. AGRON. (193)1-36, AGO, 1969
- 15 LIMA, O.G. DE MARINI-BETTOLO, G.B., MONACHE, F.D., COELHO, J.S. DE H.,
D'ALBUQUERQUE, I.L., MACIEL, G.M., LACERDA, A., MARTINS, O.G.
SUBSTANCIAS ANTI MICROBIANAS DE PLANTAS SUPERIORES, COMUNICACAO XXVII, ATIVIDADE
ANTI-MICROBIANA E ANTI NEOPLASICA DE PRODUTO IDENTIFICADO COMO 2 MET OXI 6 N
PENTIL P MENZO QUINONA (PRIMINA) ISO:ADO DE RAIZES DE MICONIA-SP. (MERB, IAUPE
1903) (MELASTOMACEAE)
B. INST. ANTIB. 10(1/2)29-34, DEZ. 1970

.....

18 REFERENCIA(S) ENCIMTRADA(S) PARA A PESQUISA "PELE"
DESSAS, APENAS 15 SERAO LISTADAS.

- 1 PSILLAKIS, J.M., JUNGE, F.B. DE ALBANO, A., SPINA, V.
AUTO ENXERTOS DE PELE, ESTUDO DO MAGNESIO, CALCIO, FOSFORO E COBRE
Q. HOSPITAL 77(1)101-5, JAN, 1970
- 2 PANDOLFO, S.M.
ENXERTO DE PELE TOTAL, APUS MASTECTOMIA RADICAL
B. GINCOL. E OHST. 126(2)63-7, FEV, 1970

- 3 PITANGUY, I., TORRES, E.T.
TUMORES DA PÉLF, NOVA TÉCNICA PARA UTILIZAÇÃO DOS CORTES DE CONGELADO
TMIH. MED. (363)15-7, JUN, 1970

- 4 WHITAKER, J.G.
ABSORÇÃO DE HORMÔNIOS E VITAMINAS PELOS CAPILARES DAS MUCOSAS E DA PELE
UTILIZAÇÃO CLÍNICA
R. BRAS. MED. 27(9)441-4, SET, 1970

- 5 NEVES, H.G., CARNEIRO, J., FRANCA, M.J.C.R.
PRESENÇA DE COGUMELOS NA CAMADA CÚRNEA DA PELE DE DOENTES E INDIVÍDUOS SUSPEITOS
DE LLPRA
B. SEVY. NAC. LEPRO 29(1/2)77-85, MAH./JUN, 1970

- 6 FRANCA, J.G.C.
EMPREGO DE ENXERTO DE PELE E DE FÁSCIA-LATA NA RECONSTRUÇÃO DE PAREDE TORÁCICA
POS MASTECTOMIA RADICAL E AMPLIADA
R. CBC 6(14)335-6, NOV./DEZ, 1970

- 7 AMARAL, R.
OS SINAIS NA PELE SE TRANSFORMAM EM CANCER
LAV. ARROZ. 23(257)46, SET./OUT, 1970

- 8 HERNANDEZ, C.D.V., GERVINI, R.L., MULLER, L.F.B.
TUMORES MALIGNOS DA PELE
R. MED. (5)467-88, 1970

- 9 CIRNE, B., MALNIC, G.
EFEITO DE ESTERÓIDES SOBRE O TRANSPORTE DE SÓDIO EM PELE DE SAPO
IN = REUNIAO ANUAL DA SBPC, XXII, SALVADOR, 1970
RESUMOS
SAO PAULO, SOCIEDADE BRASILEIRA PARA O PROGRESSO DA CIENCIA, 1970.

- 10 VIEIRA, F.L., CAPLAN, S.R., ESSIG, A.
ENERGÉTICA DO TRANSPORTE DE SÓDIO PELA PELE DE RA, I, CONDIÇÃO DE CURTO CIRCUITO
IN = REUNIAO ANUAL DA SBPC, XXII, SALVADOR, 1970
RESUMOS
SAO PAULO, SOCIEDADE BRASILEIRA PARA O PROGRESSO DA CIENCIA, 1970, p. 327.
RESUMO

- 11 VIEIRA, F.L., CAPLAN, S.R., ESSIG, A.
ENERGÉTICA DO TRANSPORTE DE SÓDIO PELA PELE DE RA, II, EFEITOS DE PERTURBAÇÕES
ELETRICAS
IN = REUNIAO ANUAL DA SBPC, XXII, SALVADOR, 1970
RESUMOS
SAO PAULO, SOCIEDADE BRASILEIRA PARA O PROGRESSO DA CIENCIA, 1970, p. 327.
RESUMO

- 12 ROCHA, G.L.
INFILTRAÇÃO LINFOCITÁRIA DA PELE
AN. BRAS. DERMATOL. 45(2)105-16, ABH./JUN, 1970

- 13 SAMPAIO, S.H., FRAGA, S.,
ESTUDO COMPARATIVO DA ESTRUTURA DA PELE EM DIFERENTES AREAS DO CORPO HUMANO
AN. BRAS. DERMATOL. 45(2)131-46, ABR./JUN, 1970

- 14 ROPEIRO, M.M.,
DADOS PARA O CURTIMENTO DOMESTICO DA PELE DE COELHO
SAO PAULO, DEPARTAMENTO DA PRODUCAO ANIMAL, 1970, 4 P.,
(SEN, VULGARIZACAO, TECNOLOGIA ANIMAL, 1)

- 15 ROPEIRO, M.M.,
FATORES QUE INFLUEM NA PRODUCAO DA PELE
SAO PAULO, DEPARTAMENTO DA PRODUCAO ANIMAL, 1970, 4 P.,
(SEN, VULGARIZACAO, CUNICULTURA, 24)

*** S I R I U S *** LISTAGEM DO PROGRAMA DE BUSCA *** INPE/DPO ***

DEMONSTRAÇÃO DO SIRIUS

SEBASTIAO PEREIRA 30/JUN/73 21,19,21 ENDEREÇO IGNORADO

OS RESULTADOS ENCONTRADOS NAS PESQUISAS ANTERIORES NÃO FORAM SA-
TISFATORIOS; TENTAMOS NOVAMENTE, COM MAIORES RESTRICÕES,

<COELHO> MAS NÃO <COELHO J> E "MEDICINA"

"PELE" = ("COELHO" OU "RA")

APESAR DE UM "SINÔNIMO" DE RA, QUE NÃO NOS INTERESSA, AS BUSCAS
PRODUZIRAM UMA COMPREENSÍVEL BIBLIOGRAFIA,

*** S I R I U S *** LISTAGEM DAS REFERENCIAS ENCONTRADAS *** INPE/OPD ***

DEMONSTRAÇÃO DO SIRIUS

SEBASTIAO PENEIRA 30/JUN/73 21:19,21 ENDEREÇO IGNORADO

10 REFERÊNCIA(S) ENCONTRADA(S) PARA A PESQUISA <COELHO> MAS NAO <COELHO J> E "MEDICINA"

- 1 BOHNER, M.S. DE A, LAUN, I, BENTO COELHO SOUZA, P.L.R.,
HIPER TIREOIDISMO; RELATO DE UM CASO COM 16 MESES
J. PEDIAT. 35(1/2)39-42, JAN./FEV, 1970
- 2 COELHO, L, ASSIS, L.M.,
ESTUDO CLINICO DO TRABALHO MENTAL DE 42 PACIENTES EPILEPTICOS ADOLESCENTES
ARQ. NEURO-PSIQ, 28(2)135-40, JUN, 1970
- 3 BATISTA FILHO, M, COELHO, M.A.L, BEGHIN, I.D.,
CENTRO DE EDUCACAO E RECUPERACAO NUTRICIONAL, PRIMEIRA EXPERIENCIA
O HOSPITAL 77(2)426-39, FEV, 1970
- 4 HERSZENWUT, E.P, FRANCO, L.G.P, COELHO, M.S.,
SOBRE UM CASO DE HEMO GLOBINURIA PAROXISTICA NOTURNA E GRAVIDEZ
J. BRAS. GINECOL, 69(2)89-100, FEV, 1970
- 5 BONOMO, I, BATISTA, R, COELHO, S.R, MARINHO, M.A.A.,
LABORATORIO EM DOENCAS REUMATICAS
J. BRAS. MED, 18(2)15-36, FEV, 1970
- 6 ALVARO, P, ALBUQUERQUE, E.C, DE TOSCANO, R.P, COELHO, A, MELO, R.F, DE A,
CORREIA, A.R, ALMEIDA, O, ARRAES, A.M, DA S,
MANIFESTACOES RARAS DA ESQUISTOSSOMOSE-MANSONICA
AN. FAC. MED, UNIV. FED. PERNAMBUCO (28/29)15-23, 1968/69
- 7 BOHNER, M.S. DE A, LAUN, I, COELHO, B, SOUSA, P.L.R.,
HIPER TIREOIDISMO NA INFANCIA, REVISAO A PROPOSITO DE UM CASO COM 16 MESES
ARQ. BRAS. ENDOCRINOL. E METABOL, 18(2)107-20, AGO, 1969
- 8 COELHO, L.F, DE S.,
ESTUDO COMPARATIVO DA DEGENERACAO HIOROPICA E O ACUULO DE GLICOGENIO NO
EPITELIO DA GENGIVA HUMANA
ARQ. CENTRO EST. FAC. ODONT, UNIV. FED, M, GERAIS 7(1)71-7, JAN./JUN, 1970
- 9 PEREIRA FILHO, J.S, COELHO, X, SILVANY FILHO, A.M, CALEGARO, J.,
CANCER DE PENIS
ARQ. ONCOL, 10(1)106-13, 1969

- 10 COELHO, C.A. DE A.
ESTASE UUDIFNAL E ULCERA PEPTICA
RIO DE JANEIRO, FACULDADE DE MEDICINA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO,
1970. 58 P. IESE

10 REFERENCIA(S) ENCONTRADA(S) PARA A PESQUISA "PELE" = ("COELH." OU "RA.")

- 1 PSILLAKIS, J.M., JONGE, F.B. DE ALBANO, A., SPINA, V.
AUXILIO ENXERTOS DE PELE, ESTUDO DO MAGNESIO, CALCIO, FOSFORO E COBRE
O HOSPITAL 77(1)101-5, JAN, 1970
- 2 PANUOLIU, S.M.
ENXERTO DE PELE TOTAL, APÓS MASTECTOMIA RADICAL
M. GINECOL. E OBST. 126(2)63-7, FEV, 1970
- 3 PITANGUY, I., TORRES, E.T.
TUMORES DA PELE, NOVA TECNICA PARA UTILIZACAO DOS CORTES DE CONGELACAO
TRIB. MED. (36)19-7, JUN, 1970
- 4 WHITAKER, J.G.
ABSORCACAO DE HORMONIOS E VITAMINAS PELOS CAPILARES DAS MUCOSAS E DA PELE
UTILIZACAO CLINICA
M. BRAS. MED. 27(9)441-4, SET, 1970
- 5 NEVES, R.G., CARNEIRO, J., FRANCA, M.J.C.B.
PRESENCA DE COGUMELOS NA CAMADA CORNEA DA PELE DE OBITOS E INDIVIDUOS SUSPEITOS
DE LEPRA
M. SERV. NAC. LEPROS. 29(1/2)77-85, MAR./JUN, 1970
- 6 FRANCA, J.G.C.
EMPREGO DE ENXERTO DE PELE E DE FASCIA-LATA NA RECONSTRUCAO DE MAMELO TORACICA
PÓS MASTECTOMIA RADICAL E AMPLIADA
R. CBC 6(14)335-6, NOV./DEZ, 1970
- 7 AMARAL, R.
OS SINAIS NA PELE SE TRANSFORMAM EM CANCER
LAV. ANOZ. 23(257)46, SET./OUT, 1970
- 8 BERNARDI, C.D.V., GERVINO, R.L., MULLER, L.F.B.
TUMORES MALIGNOS DA PELE
M. MED. (5)467-88, 1970
- 9 CILNE, H., MALINI, G.
EFITO DE ESTEROIDES SOBRE O TRANSPORTE DE SODIO EM PELE DE SAPI
IN = REUNIAO ANUAL DA SBPC, XXII, SALVADOR, 1970
RESUMOS
SAO PAULO, SOCIEDADE BRASILEIRA PARA O PROGRESSO DA CIENCIA, 1970.
- 

- 10 ROCHA, G.L.
INFILTRACAO LINFOCITARIA DA PELE
AN. BRAS. DERMATOL. 45(2)105-16, ABR./JUN, 1970
- 11 SAMPAIO, S.H. FRAGA, S.
ESTUDO COMPARATIVO DA ESTRUTURA DA PELE EM DIFERENTES AREAS DO CORPO HUMANO
AN. BRAS. DERMATOL. 45(2)131-46, ABR./JUN, 1970
- 12 ROMEIRO, M.M.
FATORES QUE INFLUEM NA PRODUCAO DA PELE
SAO PAULO, DEPARTAMENTO DA PRODUCAO ANIMAL, 1970, 4 P.
(SER. VULGARIZACAO, CUNICULTURA, 24)
- 13 FRAGA, S.
PELE E DIABETES
AN. BRAS. DERMATOL. 45(3)271-4, JUL./SET, 1970
- 14 MOREIRA, Y.K. BARBOSA, M. MOREIRA, E.C. FONSECA, I.C.
FUNGOS QUERATINOFILICOS PATOGENICOS PARA O HOMEM, NOS PELOS E PELE DE CAES E
GATOS NORMAIS
ARQ. ESC. VET. UNIV. FED. M. GERAIS (22)141-4, 1970

*
* *** S I R I U S *** LISTAGEM DO PROGRAMA DE BUSCA *** INPE/OPD *** *
*
*
*
*
*
*
*

DEMONSTRAÇÃO DO SIRIUS

SEBASTIAO PEREIRA 30/JUN/73 21,19,21 INPE, S. J. CAMPOS

AGORA MOSTRAMOS UMA BUSCA NA QUAL O USUARIO PARTE DE UM TERMO
BASTANTE GENERICO, CHEGANDO RAPIDAMENTE A UM ASSUNTO BEM ESPECIFICO,

- # A # "ANAL"
- # B # "QUIMIC" E A
- # C # "SOLO" E B

NORMALMENTE, OS DOIS PRIMEIROS COMANDOS SERIAM DO TIPO #,
COLOCANDO-OS COMO # PARA MOSTRAR A RAPIDEZ COM QUE O OPERADOR "E" REDUZ
OS CONJUNTOS.

*** S I M I U S *** LISTAGEM DAS REFERENCIAS ENCONTRADAS *** INPE/DPO ***

DEMONSTRAÇÃO DO SIRIUS

SEBASTIAO PEREIRA 30/JUN/73 21,19,21 INPE, S. J. CAMPOS

281 REFERÊNCIA(S) ENCONTRADA(S) PARA A PESQUISA A "ANAL"
DESSAS, APENAS 15 SERÃO LISTADAS.

- 1 MACHADO, L.V.
A DUR NO TRABALHO DE PARTO, PRINCÍPIOS GERAIS DA ANALGOTOCIA
GO 4(8)32-46, AGO, 1970
- 2 GRAHERT, M, SONDA, T, TELES, C, ARAUJO, R, GUIMARAES, S,
GUARNIEHE NETTO, C, DE
PNEUMO PERITONIO PELVIGRAFIA, ANALISE DE 100 CASOS, DIAGNOSTICO RADIOLOGICO DO
OVARIO POLICISTICO
R. PAUL. MED. 76(4/6)53-6, JUN, 1970
- 3 BAUER, L, SILVA, G.A, DE A,B, E
CONTRIBUICAO A ANALISE DO OLEO ESSENCIAL DE LANTANA-MONTEVIDENSISRIQUET
R. BRAS. FARM. 51(3)131-8, MAIO/JUN, 1970
- 4 BARNETO, R.C.H,
O AVICULTOR E A GOTA OU PROTEINA BRUTA, DADO ANALITICO DE VALOR OUVIDOSO
AVIC, BRAS, 7(82)36-7, OUT, 1970
- 5 SILVA, M.J.D, E
FATOR DEMOGRAFICO NO ESTADO DO PANA, ANALISE PRELIMINAR
BELEM, SUPERINTENDENCIA DO DESENVOLVIMENTO DA AMAZONIA, 1970, 23 P,
- 6 CAMPOS, J.L,
DIMENSIONAMENTO E ANALISE DE INSTABILIDADES ESTATICAS EM TROCADORES DE CALOR
IN - REUNIAO ANUAL DA SBPC, XXII, SALVADOR, 1970
RESUMOS
SAO PAULO, SOCIEDADE BRASILEIRA PARA O PROGRESSO DA CIENCIA, 1970, P, 188,
RESUMO
- 7 VENCovsky, R, DIAS, O.J, RICHARD, Y,
UM MODELO GENETICO APLICADO A ANALISE DE DADOS DE PRODUCAO DE LEITE EM GAUD
BOVINO (II)
IN - ESALQ
RELATORIO CIENTIFICO DE 1970, ESCOLA SUPERIOR DE AGRICULTURA LUIZ DE QUEIROZ
PIRACICABA, ESCOLA SUPERIOR DE AGRICULTURA LUIZ DE QUEIROZ, 1970, P, 130-6

- 8 MENEZES JUNIOR, A.A., PHACA, J.C.G.
ANALISE DE SUITOS DE TENSÃO RESULTANTES DE MANOHA NO SISTEMA DA CHESF ATRAVÉS O
CONCEITO DE VARIÁVEIS DE ESTADO, COMPANHIA HIDRO ELÉTRICA DO SÃO-FRANCISCO
IN - COMISION DE INTEGRACION ELECTRICA REGIONAL
EFECTIVIDAD EN LA PROTECCION CONTRA SOBRETENSIONES
MONTEVIDEO, COMISION DE INTEGRACION ELECTRICA REGIONAL, 1970, v. 3, p. 1-39
- 9 CASADEL, A.L.
ANALISE TÉCNICA DO COMBUSTÍVEL DE UM REATOR DE POTENCIA
SÃO PAULO, ESCOLA POLITECNICA DA UNIVERSIDADE DE SÃO PAULO, 1970, 89 F, TESE
- 10 CINTRA, S.M.L., GENTILE, E.F., NISHIOKA, I., ABRAO, M.A. DE S.,
AMBROZIO FILHO, F.
ANALISE DE VARIÁVEIS DO PROCESSO DE FABRICACAO DE PLACAS COM NUCLEOS DE
DISPENSOES AT-U308
METALURGIA 26(146)31-43, JAN, 1970
IN - REUNIAO ANUAL DA SBPC, XXII., SALVADOR, 1970
RESUMOS
SÃO PAULO, INSTITUTO DE ENERGIA ATOMICA, 1970, 13 P, RESUMO
(PUBLICACAO IEA, 203)
- 11 RAMOS, E.L.
ECONOMIA BATANA, ESTUDO SOBRE O SETOR AGRICOLA, I, PARTE, ANALISE DO SETOR
AGRICOLA DA BAHIA
AGRU-ECON, 2(4)9-50, OUT./DEZ, 1970
- 12 RUSCHI, A.
DIFERENÇAS DO ESTADO DO ESPÍRITO-SANTO, CHAVES ARTIFICIAIS E ANALITICAS PARA
DETERMINAR OS GÊNEROS DAS SUBTRIBUS DICHAEA E SARCANTHEAS E AS ESPÉCIES DOS
GÊNEROS DICHAEA, EPITHECIA E CAMPYLOCENTRUM, REPRESENTADAS NO ESTADO DO
ESPÍRITO-SANTO
B. MUS. BIOL. PROF. MELLO LEITAO, SER, BOTANICA (75)1-10, 1970
- 13 RUSCHI, A.
DIFERENÇAS DO ESTADO DO ESPÍRITO-SANTO, CHAVES ARTIFICIAIS E ANALITICAS PARA
DETERMINAR OS GÊNEROS DA SUBTRIBU LAELIEAS, REPRESENTADAS NO ESTADO DO ESPÍRITO-
SANTO
B. MUS. BIOL. PROF. MELLO LEITAO, SER, BOTANICA (46)1-10, 1970
- 14 SCHMIDT-HEBHEL, M., OLIVEIRA, J.S. DE
ANALISES DE LEITE
IN - *
CURSO DE ANALISE QUÍMICA DE ALIMENTOS
CAMPINAS, INSTITUTO DE TECNOLOGIA DE ALIMENTOS, 1970, P. 124-37
- 15 ANDRADE, F. DE
BATATA, ANALISE CONJUNTURAL
R. ECON, RASA 1(1)27-32, SET./DEZ, 1970

1
30
30
1

10 REFERENCIA(S) ENCONTRADA(S) PARA A PESQUISA B = "QUÍMICA" E A

- 1 SENISE, P.
QUÍMICA ANALÍTICA
CI. E CULT. 22(2)93-8, JUN. 1970
- 2 GUIMARAES, G. DE A., BASTOS, J.B., LOPES, E. DE C.
MÉTODOS DE ANÁLISE FÍSICA, QUÍMICA E INSTRUMENTAL DE SOLOS
BELEM, INSTITUTO DE PESQUISAS E EXPERIMENTAÇÃO AGROPECUARIAS DO NORTE, 1970, 106
P.
- 3 UMORI, L.H., ONAKA, Y.
ALGUMAS CONSIDERAÇÕES SOBRE A APLICAÇÃO DA ESPECTROFOTOMETRIA DE ABSORÇÃO
ATÔMICA NA DETERMINAÇÃO QUANTITATIVA DE ÓXIDOS DE FERRO E SILÍCIO NA ANÁLISE
QUÍMICA DE MATERIAIS REFRACTÓRIOS
CERÂMICA 16(63)231-3, JUL./SET. 1970
- 4 CEREDA, M.P., CARONARI, R.
ANÁLISES QUÍMICAS EM VARIEDADES DE MORANGOS (FRAGARIA-HÍBRIDA)
IN = REUNIAO ANUAL DA SUPC, XXII, SALVADOR, 1970
RESUMOS
SAO PAULO, SOCIEDADE BRASILEIRA PARA O PROGRESSO DA CIENCIA, 1970, P. 233-4,
RESUMO
- 5 SILVA, E.
NOVA ORIENTAÇÃO PARA APRENDIZAGEM EXPERIMENTAL DE QUÍMICA DE ACORDO COM AS
TÉCNICAS DA ANÁLISE DE TOQUE
RECIFE, UNIVERSIDADE FEDERAL DE PERNAMBUCO, 1970, 53 P.
- 6 GALLO, J.R., MIHOCE, R., BATAGLIA, O.C., MORAES, F.R.P. DE
LEVANTAMENTO DE CAFÉZAIS DO ESTADO DE SÃO PAULO PELA ANÁLISE QUÍMICA FOLIAR, II,
SOLOS PODZOLIZADOS DE LINS E MARILIA, LATOSSOLO ROXO E PODZOLICO VERMELHO
AMARELO-ORTO, 12, CONGRESSO-BRASILEIRO-DE-CIENCIA-DO-SOLO /SAO-PAULO/
BRAGANTIA 29(22)237-47, JUL. 1970
- 7 GALLO, J.R., MIHOCE, R., NOBREGA, S. DE A.
NUTRIÇÃO NITROGENADA, FOSFATADA E POTÁSSICA DA BATATINHA, REVELADA PELA ANÁLISE
QUÍMICA FOLIAR, 12, CONGRESSO-BRASILEIRO-DE-CIENCIA-DO-SOLO
BRAGANTIA (29)1-V, JAN. 1970 (NOTA N. 1)
- 8 *
CURSO DE ANÁLISE QUÍMICA DE ALIMENTOS
CAMPINAS, INSTITUTO DE TECNOLOGIA DE ALIMENTOS, 1970, 199 P.
- 9 SCHMIDT-HEBREL, M.
ANÁLISE QUÍMICA DA ÁGUA PARA A INDÚSTRIA DE ALIMENTOS
IN * *
CURSO DE ANÁLISE QUÍMICA DE ALIMENTOS
CAMPINAS, INSTITUTO DE TECNOLOGIA DE ALIMENTOS, 1970, P. 100-10

- 10 SCHMIDT-HEBBEL, H.
IMPORTANCIA DA ANALISE QUIMICA NA TECNOLOGIA DE ALIMENTOS
IN * *
CURSO DE ANALISE QUIMICA DE ALIMENTOS
CAMPINAS, INSTITUTO DE TECNOLOGIA DE ALIMENTOS, 1970, P. 1-6

2 REFERENCIA(S) ENCONTRADA(S) PARA A PESQUISA C ■ "SOLO" E B

- 1 GUIARAES, G. DE A. BASTOS, J.B. LOPES, E. DE C.
METODOS DE ANALISE FISICA, QUIMICA E INSTRUMENTAL DE SOLOS
BELEM, INSTITUTO DE PESQUISAS E EXPERIMENTACAO AGROPECUARIAS DO NORTE, 1970, 106
P.
- 2 GALLO, J.R. HIROCE, R. BATAGLIA, O.C. MORAES, F.R.P. DE
LEVANTAMENTO DE CAFEZAIS DO ESTADO DE SAO-PAULO PELA ANALISE QUIMICA FOLIAR, II,
SOLOS PODZOLIZADOS DE LINS E MARILIA, LATOSSOLO ROXO E PODZOLICO VERMELHO
AMAHEL-ORTO, 12, CONGRESSO-BRASILEIRO-DE-CIENCIA-DOSOLO /SAO-PAULO/
BRAGANTIA 29(22)237-47, JUL. 1970

*** S I R I U S *** LISTAGEM DO PROGRAMA DE BUSCA *** INPE/DPD ***

DEMONSTRAÇÃO DO SIRIUS

SEBASTIAO PEREIRA 30/JUN/73 21,19,21 INPE, S, J, CAMPOS

NESTE EXEMPLO MOSTRAMOS COMO É POSSÍVEL CHEGAR A EXPRESSÕES DE BUSCA COMPLEXAS, SEM GRANDES DIFICULDADES DE CODIFICAÇÃO.

QUEREMOS ENCONTRAR TRABALHOS MÉDICOS, QUE TRATEM DE PEIXES OU RATOS, E CUJOS AUTORES NÃO SEJAM NEM GUIARAES NEM MACHADO, ALÉM DISSO, PRETENDENDO QUE O ESTUDO, CASO SE REFIRA A RATOS, TAMBÉM MENCIONE RINS.

É FÁCIL CHEGAR À BUSCA FINAL, DEFININDO ALGUNS TERMOS AUXILIARES,

RINS = "RIM" OU "RINS"
MAGU = <MACHADO> OU <GUIARAES>
RATO = ("RATO" OU "RATUS") E RINS
FISH = "PEIXE" OU "PEIXES"

((FISH OU RATO) E "MEDICINA")=MAGU

ESTA BUSCA TAMBÉM PODE SER INTEIRAMENTE DESCRITA EM UM ÚNICO COMANDO, VALENDOS DE PREFIXOS,

("PEIX*"OU"RATO*"E("RIM*"OU"RINS*))E"MEDICINA"=(<MACHADO>OU<GUIARAES>)

DEMONSTRAÇÃO DO SIRIUS

SEBASTIAO PEREIRA 30/JUN/73 21.19.21 INPE, S. J. CAMPOS

6 REFERÊNCIA(S) ENCONTRADA(S) PARA A PESQUISA ((FISH OU RATO) E "MEDICINA")=MAGU

- 1 CORREIA, M. DE R., NUNGEL, J.M. DE T., PIZA, J. DE T., RAMOS, A. DA S., DIAS, L.C. DE S., MOHAI, L.V.C. E ROSARIO, F.F. DO
DISPERSAO DE BIOTRANSFORMAÇÃO-STRANINEA, HOSPEDEIRA INTERMEDIARIA DO SCHISTOSOMA-
MANSONI, ATRAVES DA DISTRIBUICAO DE PEIXES
R. SAUDE PUBL, 4(2)117-27, DEZ. 1970
- 2 PASSOS, D.B.,
PERFUZAO DE DIVERTICULO DE WECKEL POR ESPINHA DE PEIXE, SIMULANDO APENDICITE
AGUDA
R. BRAS. CIR. 60(3/4)203-6, SET./OUT. 1970
- 3 MELLO, G.H. DE MALNIC, G.,
ALGUMAS CARACTERISTICAS ELÉTRICO FISIOLÓGICAS DE TUBULOS PRÓXIMOS DE RINS DE RATO
IN = REUNIAO ANUAL DA SBPC, XXII, SALVADOR, 1970
RESUMOS
SAO PAULO, SOCIEDADE BRASILEIRA PARA O PROGRESSO DA CIENCIA, 1970, P. 326.
RESUMO
- 4 MALNIC, G., AIRES, M. DE M.,
MECANISMO DE ACIDIFICACAO NO TUBULO PROXIMAL DE RINS DE RATO
IN = REUNIAO ANUAL DA SBPC, XXII, SALVADOR, 1970
RESUMOS
SAO PAULO, SOCIEDADE BRASILEIRA PARA O PROGRESSO DA CIENCIA, 1970, P. 325.
RESUMO
- 5 SIQUEIRA, M.,
ESTUDOS SOBRE A ATIVIDADE NEFROTÓXICA DAS FRAÇÕES GAMA-1 E GAMA-2, DE SORO DE
CUBAIA ANTI RIM DE RATO
O BIOLOGICO 36(12)336-7, DEZ. 1970, RESUMO
- 6 WOLSKANN, T.V., FERREI, S., MEDEIROS, L.O.,
CONTRIBUICAO AO ESTUDO HISTOQUÍMICO DOS POLI SACARÍDEOS EM NEFRON DE PEIXES DE
ÁGUA-DOCE
IN = REUNIAO ANUAL DA SBPC, XXII, SALVADOR, 1970
RESUMOS
SAO PAULO, SOCIEDADE BRASILEIRA PARA O PROGRESSO DA CIENCIA, 1970, P. 311-2.
RESUMO

6 REFERENCIA(S) ENCONTRADA(S) PARA A PESQUISA ("PEIXE" OU "RATO" E ("RIM" OU "RINS")) E "MEDICINA" - (<MACHADO> OU <GUIMARAES>)

- 1 CORREA, R. DE R., MURGEL, J.M. DE T., PIZA, J. DE T., RAMOS, A. DA S., DIAS, L.C. DE S., MORAIS, L.V.C. E ROSARIO, F.F. DO DISPEPSIA DE BIOMPHALARIA-STRAMINEA, HOSPEDEIRA INTERMEDIARIA DO SCHISTOSOMA-MANSONI, ATRAVES DA DISTRIBUICAO DE PEIXES R, SAUDE PUBL, 4(2)117-27, DEZ, 1970
- 2 PASSOS, D.B., PERFURACAO DE DIVERTICULO DE MECKEL POR ESPINHA DE PEIXE, SIMULANDO APENDICITE AGUDA R, BRAS, CIR, 60(3/4)203-6, SET./OUT, 1970
- 3 MELLO, G.R. DE MALNIC, G., ALGUMAS CARACTERISTICAS ELETHO-FISIOLOGICAS DE TUBULOS PROXIMAIS DE RINS DE RATO IN - REUNIAO ANUAL DA SBPC, XXII, SALVADOR, 1970 RESUMOS SAO PAULO, SOCIEDADE BRASILEIRA PARA O PROGRESSO DA CIENCIA, 1970, P, 326, RESUMO
- 4 MALNIC, G., AIRES, M. DE M., MECANISMO DE ACIDIFICACAO NO TUBULO PROXIMAL DE RINS DE RATO IN - REUNIAO ANUAL DA SBPC, XXII, SALVADOR, 1970 RESUMOS SAO PAULO, SOCIEDADE BRASILEIRA PARA O PROGRESSO DA CIENCIA, 1970, P, 325, RESUMO
- 5 SIQUEIRA, M., ESTUDOS SOBRE A ATIVIDADE NEFROTOXICA DAS FRACOES GAMA-1 E GAMA-2, DE SORO DE COBALTA ANTI RIM DE RATO O BIOLOGICO 36(12)336-7, DEZ, 1970, RESUMO
- 6 WGRSMANN, T.V., FERRI, S., MEDEIROS, L.O., CONTRIBUICAO AO ESTUDO HISTOQUIMICO DOS POLI SACARIDEOS EM NEFRON DE PEIXES DE AGUA-DOCE IN - REUNIAO ANUAL DA SBPC, XXII, SALVADOR, 1970 RESUMOS SAO PAULO, SOCIEDADE BRASILEIRA PARA O PROGRESSO DA CIENCIA, 1970, P, 311-2, RESUMO

*** S I R I U S *** LISTAGEM DO PROGRAMA DE BUSCA *** INPE/OPD ***

DEMONSTRAÇÃO DO SIRIUS

SEBASTIAO PEREIRA 30/JUN/73 21,19,21 INPE, S. J. CAMPOS

NO ULTIMO EXEMPLO, ESTAMOS INTERESSADOS EM ARTIGOS NOS QUAIS OS AUTORES TRATAM O PROBLEMA DA DOR NO PARTO, ESTE EXEMPLO MOSTRA, TAMBEM, A NECESSIDADE DE RECONHECER A NOMES DE CAMPOS CORRELATOS PARA PREENCHER UMA DADA NECESSIDADE DE BUSCA.

- # 1 # "PARTO" OU "PARTURIENTE" OU "GINECO" OU "OBSTETRI"
- # 2 # "DOR" OU "DORES" OU "ANAL" = "ANALISE" OU "ANESTES"

- # 1 E 2 E "MEDICINA"

*** S I R I U S *** LISTAGEM DAS REFERÊNCIAS ENCONTRADAS *** INPE/OPD ***

DEMONSTRAÇÃO DO SIRIUS

SPRÁSTIAD PEREIRA 30/JUN/77 21.19.21 INPE, S. J. CAMPOS

9 REFERÊNCIA(S) ENCONTRADA(S) PARA A PESQUISA 1 E 2 E "MÉDICA"

- 1 MALHE, M.M., FREITAS, C.A., DE SIQUEIRA, C.A., PÉTI, D.A., ANDRADE, F.S.M., OLIVEIRA, J., PAES, J.A.F.R., FERREIRO, L.M., ABRÃO, N., DONADIO, N., MACHADO, R.A., SAKAYA, R.B.
ANALGESIA OBTETRICA; COMPARAÇÃO ENTRE OS EFEITOS DO FLOMOGLUCINOL E DA ASSOCIAÇÃO DIAZEPAM PENTAZOCINA EM PARTOS TRANS VAGINAIS
J. BRAS. GINECOL. 70(11)14, JUL, 1970
- 2 LAHUNIE, G.M., RIBEIRO, R.C.
MET OXI FLORANO EM AUTO ADMINISTRAÇÃO PARA ANALGESIA OBTETRICA
R. BRAS. ANESTESIOL. 20(2)206-11, ABR./JUN, 1970
- 3 MARTINS, A.D., ALTAFF, G.S.
ASSOCIAÇÃO ENZIMÁTICO ANALGESICA EM CIRURGIA GINECOLÓGICA E OBTETRICA
O HOSPITAL 77(A)2090-9, JUN, 1970
- 4 MACHADO, L.V.
A DOR NO TRABALHO DE PARTO, PRINCÍPIOS GERAIS DA ANALGOTOCIA
GO 4(8)32-46, AGO, 1970
- 5 FASSA, L.
ANALGESIA OBTETRICA, EFEITO DA ASSOCIAÇÃO FLOMOGLUCINOL TRI MET UJI BIZIENO NO TRABALHO DO PARTO
J. BRAS. GINECOL. 70(6)331-4, DEZ., 1970
- 6 MONTELEONE, P.P.R.
ESTUDO NA AÇÃO ANALGESICA, ANTI ESPASMÓDICA E TRANQUILIZANTE DO MAFANDRROL DURANTE O TRABALHO DE PARTO
O HOSPITAL 78(3)813-21, SET, 1970
- 7 MACHADO, J.A.F.
EMBULIA POR LÍQUIDO AMNÍOTICO, REVISÃO E APRESENTAÇÃO DE UM CASO DURANTE ANESTESIA GERAL PARA PARTO CESARÃO
R. BRAS. ANESTESIOL. 20(4)577-82, OUT./DEZ, 1970
- 8 GOES, S.J., FORSECA, A., MASSAOKA, D., GRABERT, M., LIMA, G.R. DE
O EMPREGO DE UM NOVO ANALGESICO ESPASMOLITICO NO POS OPERATÓRIO GINECOLÓGICO
O HOSPITAL 78(4)1137-9, OUT, 1970

- 9 BAFUTTO, M.G. MOURA, M.T. DE
EMPREGO DE UM DERIVADO BENZO DIAZEPINICO EM ANALGESIA DO PARTO
R, GOIANA MED, 16(3/4)247-52, JUL./DEZ, 1970

APÊNDICE C

ERROS DETETADOS POR SIRIUS E AS MENSAGENS CORRESPONDENTES

Ao escrever comandos de busca, é comum a ocorrência de omissões e enganos que resultam em erros de sintaxe ou em falta de recuperação. O sistema informa o usuário da ocorrência de erros através de mensagens na listagem do programa de busca. Em geral estas mensagens vem precedidas de uma indicação da posição do comando fonte, de busca, onde o êrro ocorreu. Estas mensagens seguem imediatamente o comando em êrro.

Hã dois tipos de mensagens: erros e avisos. No primeiro caso não é executada nenhuma busca; no segundo, o sistema toma uma atitude, arbitrãria, sôbre a causa do êrro e prossegue na busca.

A seguir discutimos todos os tipos possíveis de êrro, no mesmo formato do apêndice B.

5) O USUARIO E INFORMADO SEMPRE QUE UMA DE SUAS EXPRESSOES FOR VAZIA, ISTO E, NAO CORRESPONDER A NENHUMA REFERENCIA BIBLIOGRAFICA,

"RATAZANAS" E "CANPO"

AVISO ** >>>>> *
EXPRESSAO VAZIA, NENHUMA REFERENCIA ENCONTRADA,

H3BT = "LIXO" E "SUJO"

AVISO ** >>>>> *
EXPRESSAO VAZIA, NENHUMA REFERENCIA ENCONTRADA,

6) A AUSENCIA DO CARACTER DE ENCERRAMENTO DE PALAVRA CHAVE (OU NOME DE AUTOR) FAZ COM QUE O SISTEMA CONSIDERE COMO TERMO CHAVE TUDO O QUE HOUVER ENTRE O CARACTER DE ABERTURA E O FIM DO COMANDO,

A = "MAQUINA"
"ELETRICA E A]

AVISO ** >>>>> *
TERMO=CHAVE NAO TERMINADO, = ADMITIDO PRESENTE,

AVISO ** >>>>> *
EXPRESSAO VAZIA, NENHUMA REFERENCIA ENCONTRADA,

<GUINHARAES

AVISO ** >>>>> *
TERMO=CHAVE NAO TERMINADO, > ADMITIDO PRESENTE,

7) TODO COMANDO DE DEFINICAO IMPLICA NA PRESENCA DE UM TERMO AUXILIAR, O CARACTER "=" E UMA EXPRESSAO DE BUSCA,

JACA = "JAG*" OU "JAC"
JACA / "CAJU*"

ERRR ** >>>>> *
EXPRESSAO NAO FOI PRECEDIDA DE TERMO AUXILIAR,

8) O TERMO AUXILIAR DEFINIDO NAO PODE SER REDEFINIDO, OU SEJA, APARECER NOVAMENTE A ESQUERDA DO CARACTER "=",

ERG = "ENERGIA"
ERG = "POTENCIA"

AVISO ** REDEFINICAO DO TERMO ERG , IGNORADA,

APÊNDICE D

SINTAXE DA LINGUAGEM DE BUSCA EM NOTAÇÃO BACKUS-NAUR

Símbolos Metalinguísticos.

São usados os seguintes símbolos metalinguísticos neste apêndice (referência 17):

- < > Sinais de menor e maior limitando um ou mais dígitos ou letras representam uma variável metalinguística cuja definição é dada por uma fórmula metalinguística.

- ::= O símbolo ::= significa "é definido como". A variável metalinguística à esquerda deste símbolo é definida pela fórmula metalinguística à sua direita.

- | O símbolo | significa "OU". Ele separa definições alternativas de uma variável metalinguística.

- { } Os símbolos { e } são usados para delimitarem expressões em linguagem descritiva, quando são necessárias ou convenientes para definir uma variável metalinguística.

Fórmulas Metalinguísticas.

Símbolos metalinguísticos são combinados de maneira a formar uma fórmula metalinguística. Uma fórmula metalinguística é uma regra que irá produzir uma sequência sintaticamente correta de caracteres e ou

símbolos. Um sinal ou símbolo numa fórmula metalinguística que não é um dos símbolos metalinguísticos, denota ele próprio. A justaposição de variáveis metalinguísticas e/ou símbolos numa fórmula metalinguística denota a justaposição daqueles elementos na construção definida.

Exemplo de uma fórmula metalinguística.

$$\langle \text{identificador} \rangle ::= \langle \text{letra} \rangle \mid \langle \text{identificador} \rangle \langle \text{letra} \rangle \mid \langle \text{identificador} \rangle \langle \text{dígito} \rangle$$

A qual é lida como:

Um <identificador> é definido como uma <letra>, ou como um <identificador> seguido de uma <letra> ou como um <identificador> seguido de um <dígito>.

A fórmula metalinguística dada acima define uma relação recursiva que é a maneira pela qual um <identificador> pode ser construído. A avaliação da fórmula mostra que um <identificador> começa por uma <letra>. A letra pode permanecer sozinha ou pode ser seguida por uma sequência de <letra> e <dígito>.

Conjunto de Caracteres.

$$\langle \text{letra} \rangle ::= A|B|C|D|E|F|G|H|I|J|K|L|M|N|O|P|Q|R|S|T|U|V|W|X|Y|Z$$
$$\langle \text{dígito} \rangle ::= 0|1|2|3|4|5|6|7|8|9$$
$$\langle \text{caracter especial} \rangle ::= .|,|:|;|{|}|_|+|-|*|/|=|%|&|#|@|$\$$$
$$\langle \text{espaço simples} \rangle ::= \{ \text{uma posição branca horizontal} \}$$
$$\langle \text{espaço} \rangle ::= \langle \text{espaço simples} \rangle \mid \langle \text{espaço} \rangle \langle \text{espaço simples} \rangle$$
$$\langle \text{caracter de cordão} \rangle ::= \langle \text{letra} \rangle \mid \langle \text{dígito} \rangle \mid \langle \text{caracter especial} \rangle \mid \langle \text{espaço simples} \rangle$$

<extremo de cordão> ::= "<extremo esquerdo> | <extremo direito>

<extremo esquerdo> ::= {ē o símbolo <}

<extremo direito> ::= {ē o símbolo >}

<caracter> ::= <caracter de cordão> | <extremo de cordão>

Símbolos Básicos.

<símbolo básico> ::= <letra> | <dígito> | <delimitador> | <vazio>

<vazio> ::= {cordão vazio de símbolos}

<delimitador> ::= <operador> | <separador> | <parênteses>

<operador> ::= <operador de busca> | <operador de transferência>

<operador de busca> ::= &|/|-|E|OU|MAS <espaço> NAO

<operador de transferência> ::= =

<separador> ::= ,|<espaço>

<parênteses> ::= <parêntese esquerdo> | <parêntese direito>

<parêntese esquerdo> ::= ([

<parêntese direito> ::=)]

Componentes Básicos.

<componente básico> ::= <termo auxiliar> | <cordão>

<termo auxiliar> ::= <letra> | <dígito> | <termo auxiliar> <letra>
| <termo auxiliar> <dígito>

<cordão> ::= <cordão normal> | <cordão completo> | <cordão sem vírgula>

<cordão normal> ::= <vazio> | <cordão normal> <caracter de cordão>

<expressões de busca> ::= <busca> | <expressões de busca> ; <busca>

<busca> ::= <expressão> | <definição>

<definição> ::= <vazio> | <termo auxiliar> <operador de transferência>
 <expressão>

<expressão> ::= <termo> | <expressão> <operador OU> <termo>

<termo> ::= <fator> | <termo> <operador E> <fator>

<fator> ::= <primário> | <fator> <operador MAS NAO> <primário>

<primário> ::= <termo auxiliar> | <termo chave>

 | <parênteses esquerdo> <expressão> <parênteses direito>

<termo chave> ::= "<termo de busca>"

 | <extremo esquerdo> <termo de busca> <extremo direito>

<termo de busca> ::= <cordão normal> | <cordão normal>*

<operador OU> ::= /|OU

<operador E> ::= &|E

<operador MAS NAO> ::= -|MAS <espaço> NAO

APÊNDICE E

PROPRIEDADES DA OPERAÇÃO MAS NAO

Sejam três conjuntos distintos A, B e C, não vazios.

A operação MAS NAO, indicada pelo operador -, pode ser definida como:

$$A-B = A \cdot \bar{B}$$

A operação MAS NAO não é comutativa. De fato, se fosse, teríamos: $A-B = B-A$. Multiplicando (E) ambos os lados por $A-B$, temos:

$$(A-B)(A-B) = (B-A)(A-B)$$

$$(A-B) = (B \cdot \bar{A})(A \cdot \bar{B})$$

$$(A-B) = (A \cdot \bar{A})(B \cdot \bar{B})$$

$$A\bar{B} = \phi$$

Ora $A-B = \phi$ somente se $A=B$, o que contraria a hipótese. Portanto $A-B \neq B-A$, em geral.

A operação MAS NAO não é associativa, isto é:

$$(A-B)-C \neq A-(B-C)$$

De fato, desenvolvendo separadamente cada um dos membros, temos:

$$(A-B)-C = (A-B) \cdot \bar{C} = (A\bar{B})\bar{C} = A\bar{B}\bar{C}$$

$$(A-B)-C = A\bar{B}\bar{C} \quad (1)$$

$A-(B-C) = A-(B\bar{C}) = A.\overline{B\bar{C}} = A\bar{B} + AC$ onde o sinal + indica OU. Multiplicando o primeiro termo do segundo membro por $(C+\bar{C})$, a expressão não se altera; temos então

$$\begin{aligned} A-(B-C) &= A\bar{B}(C+\bar{C}) + AC \\ &= A\bar{B}C + A\bar{B}\bar{C} + AC \\ &= A\bar{B}\bar{C} + AC + A\bar{B}C \\ &= A\bar{B}\bar{C} + AC \end{aligned}$$
$$A-(B-C) = A\bar{B}\bar{C} + AC \quad (2)$$

Comparando as expressões (1) e (2) temos:

$$A-(B-C) = (A-B)-C + AC$$

Como, em geral, $AC \neq \phi$, segue-se que a operação MAS NAO não é comutativa.