

Concepção e Análise de Algoritmos

LeGrep

Turma 2 - Grupo C

José Peixoto	200603103	ei12134@fe.up.pt
Pedro Moura	201306843	up201306843@fe.up.pt

9 de Maio de 2015

Conteúdo

1	Formalização do problema	2
1.1	Dados de entrada	2
1.2	Limites e condições de aplicação	2
1.3	Situações de contorno	2
1.4	Resultados esperados	2
2	Descrição da solução	2
2.1	Algoritmos de pesquisa exacta em strings	2
2.1.1	Naive	2
2.1.2	Baseado em autómato finito	2
2.1.3	Knuth-Morris-Pratt	3
3	Lista de casos de utilização	3
4	Relato das principais dificuldades encontradas no desenvolvimento do trabalho	3
5	Indicação do esforço dedicado por cada elemento do grupo	3
5.1	José Peixoto	3
5.2	Pedro Moura	3

Resumo

No âmbito da unidade curricular de Concepção e Análise de Algoritmos, foi-nos proposto o desenvolvimento de uma aplicação que emulasse as funcionalidades do utilitário da linha de comandos **grep** (globally search a regular expression and print), explorando o conhecimento acerca de algoritmos para pesquisa em strings.

1 Formalização do problema

1.1 Dados de entrada

1.2 Limites e condições de aplicação

1.3 Situações de contorno

1.4 Resultados esperados

2 Descrição da solução

2.1 Algoritmos de pesquisa exacta em strings

Na pesquisa de uma correspondência exacta de um dado padrão P num texto T , recorre-se a três algoritmos diferentes, escolhidos de forma opcional pelo utilizador:

2.1.1 Naive

Para um padrão de tamanho m e texto de tamanho n , encontra todos os deslocamentos s válidos, verificando desde o início do padrão a condição $P[1..m] = T[s + 1..m]$ para cada um dos $n - m + 1$ valores possíveis de s . Este procedimento implica, no pior dos casos, uma complexidade temporal de $\Theta((n - m + 1)m)$. Torna-se bastante ineficiente para padrões com tamanho grande, uma vez que se faz uma verificação de equivalência do tamanho do padrão a cada novo deslocamento.

2.1.2 Baseado em autómato finito

Cada carácter do texto é verificado apenas uma vez. A análise ao texto é feita em tempo constante $|T|$, porque se recorre à modelação de um autómato finito com um conjunto de estados Q de tamanho $|P|$. A função de transição δ possibilita a atribuição de novos estados q a cada novo carácter a lido do texto.

$$\delta(q, a) = \sigma(P_q a) \tag{1}$$

No ciclo principal verifica-se a cada novo carácter lido do texto se se atingiu o estado de aceitação e por conseguinte, uma correspondência exacta do padrão no texto.

A construção do autómato é uma primeira fase de processamento e envolve a determinação do alfabeto Σ do padrão. Na determinação da função de transição recorre-se a uma função auxiliar para o sufixo $\sigma(x)$ que retorna, para um dado conjunto de caracteres do alfabeto, Σ^* o maior comprimento (estado) que seja prefixo do padrão.

$$\sigma(x) = \max\{k : P_k \sqsubset x\} \quad (2)$$

2.1.3 Knuth-Morris-Pratt

3 Lista de casos de utilização

- Multi-plataforma, testado em sistemas Linux e Windows.

4 Relato das principais dificuldades encontradas no desenvolvimento do trabalho

5 Indicação do esforço dedicado por cada elemento do grupo

5.1 José Peixoto

- Tratamento dos dados de entrada.
- Mostra dos comandos de ajuda.
- Concepção dos algoritmos “naive” e baseado em autómato finito.
- Autor do relatório.

5.2 Pedro Moura

- Concepção do algoritmo Knuth-Morris-Pratt.

Referências

- [1] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, *Introduction to algorithms Third Edition*, Cambridge, MA [etc.] : The MIT Press, cop. 2009.

N.B. Este relatório não foi escrito ao abrigo do novo Acordo Ortográfico.