MPEI 2024/25 - PL 6

Algoritmos Probabilísticos: Bloom Filters

Palavras chave: Geração de strings aleatórias, Funções de dispersão (*hash functions*), Bloom Filters.

6.1 Geração aleatória de chaves

1. Crie uma função para gerar um conjunto de chaves constituídas por caracteres, todas diferentes. O comprimento de cada chave deve ser escolhido aleatoriamente (distribuição uniforme) entre i_{min} e i_{max}.

   A função deve ter como parâmetros de entrada o número de chaves a gerar (N), i_{min} e i_{max}, um vector com os caracteres a usar nas chaves e um vector com as probabilidades de cada um dos caracteres a utilizar. Se a função for chamada sem o último parâmetro, deve considerar os caracteres equiprováveis (ver a documentação da função `nargin`).

   A função deve devolver um *cell array* com o conjunto de chaves geradas.

2. Usando a função, gere um conjunto de $N = 10^5$ chaves usando todas as letras maiúsculas e minúsculas ('A' a 'Z' e 'a' a 'z') com igual probabilidade e em que $i_{min} = 6$ e $i_{max} = 20$.

3. (TPC) Também usando a função, gere um conjunto de $N = 10^5$ chaves usando todas as letras minúsculas ('a' a 'z') com as probabilidades contidas no ficheiro `prob_pt.txt` [1].

   Considere novamente $i_{min} = 6$ e $i_{max} = 20$.

6.2 Funções de dispersão

1. Considere a função Matlab `string2hash()` [2] que implementa duas funções de dispersão diferentes.

   Utilizando separadamente cada uma destas funções de dispersão, simule a inserção das chaves criadas em 6.1 em 3 *Chaining Hash Tables*, uma de tamanho 5×10^5, outra de tamanho 10^6 e a terceira de tamanho 2×10^6. Para cada uma das simulações:

   (a) Guarde um vetor com os *hashcodes* obtidos.

   (b) Registe o número de atribuições a cada uma das posições de cada *Hash Table*.

   (c) Calcule o número de colisões (em cada *Hash Table* e para cada função de dispersão).

2. Utilizando a informação obtida no exercício anterior, compare o desempenho das funções de dispersão para cada tamanho diferente da *Hash Table*, relativamente a:

   (a) Uniformidade, visualizando os histogramas dos *hascodes* com 100 intervalos;

   (b) Número de colisões e número máximo de atribuições numa mesma posição da *Hash Table*.

[1] Frequências das letras em Português (`https://pt.wikipedia.org/wiki/Frequ%C3%AAncia_de_letras`).
[2] `https://www.mathworks.com/matlabcentral/fileexchange/27940-string2hash`

6.3 Filtros de Bloom

Crie um conjunto de funções Matlab que implementem as funcionalidades de um *Bloom Filter* básico. As funções devem ter os parâmetros necessários para que seja possível criar *Bloom Filters* de diferentes tamanhos (n) e a utilização de diferentes números de funções de dispersão (k).

Na criação das diferentes funções de dispersão, adote o terceiro método descrito no slide "Como ter n funções de dispersão ?" da apresentação TP sobre funções de dispersão[3] com a função que considera ter tido o melhor desempenho na experiências que efetuou na secção 6.2.

Sugestão: Criar pelo menos 3 funções: uma para inicializar a estrutura de dados; outra para inserir um elemento (ou elementos) no filtro; uma terceira para verificar se um elemento pertence ao conjunto.

1. Com as funções que desenvolveu, crie um *Bloom Filter* para guardar um conjunto, U_1, de 1000 palavras diferentes[4]. Use um *Bloom Filter* de tamanho $n = 8000$ e $k = 3$ funções de dispersão.

2. Teste o *Bloom Filter* criado anteriormente, verificando a pertença de todas as palavras do conjunto U_1. Obteve algum falso negativo?

3. Teste o *Bloom Filter* criado anteriormente, verificando a pertença de um novo conjunto, U_2, com 100000 palavras todas diferentes das de U_1. Indique a percentagem de falsos positivos obtidos.

4. Compare a percentagem de falsos positivos obtida anteriormente com a estimativa que aprendeu nas TPs.

5. Repita os exercícios 1 e 3 para um número de funções de dispersão k de 4 até 10. Faça um gráfico com a percentagem de falsos positivos em função de k. Analisando os resultados, qual o valor ótimo k? Compare este valor com o valor teórico que aprendeu nas TPs.

[3] `https://elearning.ua.pt/mod/resource/view.php?id=1452582`

[4] Sugestão: Pode usar chaves geradas aleatoriamente ou utilizar uma lista de palavras válidas para a língua portuguesa.