
Projeto 2 - Word Wrap

Alunos:

Bárbara Darques Barros Nº 7243081
Laís Helena Chiachio de Miranda Nº 8517032

Prof.

Dr. Gustavo Enrique de Almeida Prado Alves
Batista

São Carlos, SP
8 de Dezembro de 2016

1 Introdução

Este trabalho trata o desenvolvimento de um programa que realiza a quebra automática de linhas (Word Wrap), através da qual é possível alcançar a melhor distribuição de espaços em branco entre as linhas.

2 A Função de Recorrência

Sendo $opt[j]$ o custo mínimo de se terminar um texto em uma palavra j , temos que a relação de recorrência deste problema é:

$$opt[j] = \min \left\{ opt[i - 1] + costLine(i, j), \quad 1 \leq i \leq j \right\}. \quad (1)$$

No caso, $costLine(i, j)$ é o custo da linha que inicia na palavra i e termina em j . Caso a última palavra do texto seja j , é possível garantir que a última palavra da última linha seja j , assim, basta descobrir onde essa última linha começa: a palavra i .

Os candidatos a i são todas as palavras que vêm antes de j , ou ele próprio, uma vez que é possível ter uma linha formada por apenas uma palavra. O candidato *ótimo* será a palavra que minimizará a soma do custo da última linha (= $costLine(i, j)$) mais o custo de um texto que termina em $(i-1)$.

3 Complexidade

Para preencher o vetor de estados ótimos opt , é necessário fazer combinações 2 a 2 entre cada palavra e suas antecessoras, tendo-se $O(n^2)$ em tempo:

$$n * \frac{n + 1}{2} = O(n^2)$$

onde $(n+1)/2$ é a média de verificações pra cada elemento.

Já o custo de espaço é $O(n)$, pois o vetor opt tem tamanho $(n + 1)$, no qual n é o número de palavras e o $+1$ porque foi estabelecido que a palavra inicial terá índice 1. Utiliza-se também do vetor $words$ de tamanho $(n+1)$, resultando em $O(n)$.

4 Código

Como a relação de recorrência sempre referencia índices inferiores ao da chamada corrente, foi implementada uma solução iterativa que preenche o vetor opt em ordem crescente. Para cada palavra j a ser analisada, é verificado o custo que essa impõe na linha ($costLine(i, j)$) mais a solução ótima anterior ($opt[i-1]$). Caso essa soma resulte em um valor maior que $opt[j]$ (solução ótima da palavra atual), opt receberá o valor da soma (guardado na variável new_total).

A fim de impossibilitar a escolha de configurações impraticáveis, um custo muito alto(INF) é atribuído às linhas que extrapolam a largura máxima permitida. Caso a linha formada seja válida, o valor do custo é o estipulado na descrição do trabalho: (*espaço não preenchido*)² através da função $costLine(int i, int j)$.

Para auxiliar na impressão dos resultados, um vetor $parent$ é preenchido durante os testes de otimalidade, no qual $parent[j] = i$ significa que a linha ótima que termina i começa em j . Esse vetor percorrido de trás para frente retorna a divisão ótima de linhas do texto.