

Prova de Introdução à Programação (Eng.Comp.)
Prof Alexandre Mota – 23/05/2018

Questão. Faça um programa em C de caça-palavras, modularizando-o em termos de funções auxiliares à **main**. Seu programa deve ler uma matriz quadrada de caracteres do arquivo **matriz.txt**, caractere por caractere (Estes caracteres estarão armazenados como um vetor de caracteres de tamanho indefinido¹ e sem **\n**). Em seguida, seu programa deve ler as palavras (cada uma com no máximo 10 caracteres) a serem pesquisadas a partir do arquivo **palavras.txt**, de quantidade indefinida². Neste ponto, seu programa deve iniciar a busca pelas palavras, uma a uma. Para cada palavra localizada, seu programa deve armazenar em um vetor dinâmico de estruturas a posição (linha, coluna), a orientação (direita, esquerda, cima, baixo, diagonal direita cima, diagonal direita baixo, diagonal esquerda cima e diagonal esquerda baixo) e a palavra encontrada. Ao final das buscas, seu programa deverá armazenar no arquivo **onde.bin** este vetor de estruturas.

Seu programa deverá ter pelo menos as seguintes funções:

1. `char *leVetorCaracArq(int *qtdL)`: Lê o arquivo **matriz.txt**, caractere a caractere, e armazena em um vetor dinâmico, o qual será retornado por esta função. A quantidade de caracteres lidos deve ser armazenada em `qtdL`;
2. `char **vetor2Matriz(char *vetorC, int *NN)`: converte `vetorC`, de tamanho $NN=N*N$ (ou N^2) caracteres, em uma matriz quadrada $N \times N$ (matriz como ponteiro duplo). No término da função, altera o valor de `NN` para ser `N` (pode usar `math.h`);
3. `void vetorPalavras(char *vetorP[11], int *qtdP)`: Lê o arquivo **palavras.txt** e preenche `vetorP` dinamicamente com as palavras extraídas de `arqPalavras`, uma a uma, informando a quantidade de palavras em `qtdP`;
4. `void pesquisaPalavra(char **matrizC, int N, char *vetorP[11], LocalP *localizadas, int *qtdL)`: pesquisa em `matrizC` de tamanho $N \times N$ por cada uma das palavras de `vetorP`, armazenando no vetor de estruturas localizadas e atualizando a quantidade de palavras localizadas em `qtdL` (na chamada, `qtdL` terá o tamanho de `vetorP`);
5. `void salvaLocalizadas(LocalP *localizadas, int qtdL)`: salva o vetor dinâmico localizadas (de tamanho `qtdL`) no arquivo **onde.bin**.

Seu programa **main** deverá se parecer mais ou menos com a seguinte sequência de chamadas de funções:

```
...
vet = leVetorCaracArq(&t);
...
mat = vetor2Matriz(vet, &t);
...
vetorPalavras(vetP, &qP);
...
pesquisaPalavra(mat, t, vetP, localizadas, &qP);
...
salvaLocalizadas(localizadas, qP);
...
```

Não esqueça de liberar todas as regiões de memória alocadas.

Boa sorte!!!

¹ Use `while((ch=fgetc(arq))!= EOF) {...}` para saber se chegou ao final do arquivo

² Use `while(!feof(arq)) {...}` para saber se chegou ao final do arquivo