

```
/**
 * Arquivo: scanner_encapsulado.c
 * Versão : 1.0
 * Data   : 2024-10-17 23:48
 * -----
 * Este arquivo implementa um programa cliente para o teste da interface
 * scanner.h, que "lê" um texto agrupando as letras para formar palavras
 * (tokens) que são reconhecidas como uma unidade coerente. O scanner divide
 * a string do texto em seus tokens componentes.
 *
 * Baseado em: The Art and Science of C, de Eric S. Roberts.
 *             Capítulo 10: Modular Development (pg. ).
 *
 * Prof.: Abrantes Araújo Silva Filho (Computação Raiz)
 *       www.computacaoraiz.com.br
 *       www.youtube.com.br/computacaoraiz
 *       github.com/computacaoraiz
 *       twitter.com/ComputacaoRaiz
 *       www.linkedin.com/company/computacaoraiz
 *       www.abrantes.pro.br
 *       github.com/abrantesasf
 */

#include <ctype.h>
#include "genlib.h"
#include "scanner.h"
#include "simpio.h"
#include "strlib.h"

/**** Função Main: ****/

int main (void)
{
    // Frase sem acentuação, nosso scanner não lida bem com isso!
    string frase0 = "Este e um teste!";

    // Iniciar scanner para ler a frase:
    iniciar_scanner(frase0);

    // Imprimir os tokens:
    printf("Frase 0: %s\n", frase0);
    printf("  Token: %s\n", obter_proximo_token());
    printf("  Token: %s\n", obter_proximo_token());
    printf("  Token: %s\n", obter_proximo_token());
    printf("  Token: %s\n", obter_proximo_token());
    printf("  Token: %s\n", obter_proximo_token());
    printf("  Token: %s\n", obter_proximo_token());
    printf("  Token: %s\n", obter_proximo_token());
    printf("  Token: %s\n", obter_proximo_token());
    printf("\n");

    string frase1 = "o rato roeu a roupa do rei de roma.";
    string frase2 = "THE DESIGN OF THE UNIX OPERATING SYSTEM.";

    iniciar_scanner(frase1);
    printf("Frase 1: %s\n", frase1);
    printf("  Token: %s\n", obter_proximo_token());
    printf("  Token: %s\n", obter_proximo_token());
    printf("  Token: %s\n", obter_proximo_token());
    printf("\n");
}
```

```
    iniciar_scanner(frase2);
    printf("Frase 2: %s\n", frase2);
    printf("  Token: %s\n", obter_proximo_token());
    printf("  Token: %s\n", obter_proximo_token());
    printf("  Token: %s\n", obter_proximo_token());
    printf("\n");

    printf("Frase 1: %s\n", frase1);
    printf("  Token: %s\n", obter_proximo_token());
    printf("  Token: %s\n", obter_proximo_token());
    printf("  Token: %s\n", obter_proximo_token());
    printf("\n");
}
```