

```

/*
 * Programa: regex-1
 *
 * Sintaxe: regex-1 regex
 *
 * (c) Artur Pereira, 2005
 *
 * Descrição:
 * =====
 * Recebe uma expressão regular como argumento único na linha de comando
 * que processa antes de entrar em modo iterativo para pedir uma palavra e
 * verificar se pertence ou não à linguagem descrita pela expressão regular
 * fornecida.
 */

#include <stdio.h>
#include <stdlib.h>
#include <regex.h>

int main(int argc, char *argv[ ])
{
    /* valida número de parâmetros */
    if (argc != 2)
    {
        fprintf(stderr, "\e[41mNúmero de argumentos errado\e[0m\n"
            " \e[44mSintaxe:\e[0m regex-1 regex\n");
        return EXIT_FAILURE;
    }

    /* compilação da expressão regular dada */
    regex_t preg;
    int ret;
    int cflags = REG_EXTENDED | REG_NOSUB;
    if ((ret = regcomp(&preg, argv[1], cflags )) != 0)
    {
        char errMsg[128];
        regerror(ret, &preg, errMsg, 128);
        fprintf(stderr, "Erro %d de compilação da ER: %s\n", ret, errMsg);
        return EXIT_FAILURE;
    }

    /* modo iterativo de pedido de palavra, verificação e resposta */
    char u[128];
    while (scanf("%s", u) == 1)
    {
        ret = regexec(&preg, u, 0, 0, 0);
        printf("%s --> %s\n", u, ret==0 ? "yes" : "no");
    }
    printf("God bye\n");
    return 0;
}

```