

Descrição de Requisito Funcional

Identificação	RF001
Nome	Analisa ocorrências de identificadores
Atores	Usuário
Pré-condições	Existe um arquivo-texto de entrada com palavras e sinais ortográficos.
Pós-condições	Arquivo-texto de saída com as ocorrências é gerado.
Descrição	A partir de um arquivo-texto de entrada gera como saída outro arquivo-texto que contenha os identificadores, e linhas em que cada identificador ocorre.
Fluxos de Eventos	
Fluxo Principal de Eventos	
Ação do Ator	Resposta do Sistema
1. Usuário digita o nome do arquivo texto de entrada.	2. Acessa o arquivo-texto de entrada.
	3. Analisa as ocorrências de identificadores do texto.
	4. Para cada nova ocorrência de identificador gera uma linha no formato: identificador;linha1,linha2
	4. Gera arquivo-texto (com o mesmo nome do arquivo de entrada + sufixo "-xref" e extensão ".txt") com as ocorrências dos identificadores encontrados.
	5. Apresenta a mensagem final.
Fluxos alternativos	
Fluxo alternativo: Arquivo de entrada indicado como parâmetro	
Ação do Ator	Resposta do Sistema
1. O nome do arquivo-texto é enviado como parâmetro na execução do sistema.	
Fluxo alternativo: Arquivo de entrada inexistente	
Ação do Ator	Resposta do Sistema
	1.1 Apresenta a mensagem "ERRO: arquivo de entrada inexistente."
Fluxo alternativo: Arquivo de saída inválido	
Ação do Ator	Resposta do Sistema

	3.1 Apresenta a mensagem “ERRO: na geração do arquivo de saída.”
--	--

Esboços de interface com o usuário:

a) Tela inicial:

Por favor, digite o nome do arquivo-texto de entrada: _

b) Mensagem final:

Análise realizada. Arquivo XXXXX-xref.txt gerado.

Observações

- a) Na análise de identificadores, considerar que um identificador pode possuir apenas letras, dígitos e sublinhado ('_'); os demais símbolos devem ser ignorados. Considerar também que os identificadores são 'case sensitive' (ex.: TESTE, teste, Teste – devem ser considerados identificadores diferentes)

a.1) Exemplo de texto:

```
#include <stdio.h>
#include <math.h>

int v_1;

void main() {
    printf("Hello, %d", v_1);
}
```

a.2) Identificadores encontrados:

```
include
```

```
stdio
h
include
math
h
int
v_1
void
main
printf
Hello
d
v_1
```

b) O formato do arquivo texto de saída deve ser:

```
identificador;linhas
```

Cada linha terá uma palavra, seguida do separador ponto-e-vírgula, e os números de linha (separador por vírgulas) em que o identificador aparecia no texto original.

b.1) Exemplo:

```
include;1,2
stdio;1
h;1,2
math;2
int;4
v_1;4,7
void;6
main;6
printf;7
Hello;7
d;7
```