

1 - ARQUIVO DE ENTRADA CONTENDO A GLUD

Para inserir o arquivo de entrada contendo a gramática linear unitária à direita, o usuário deve **seguir de forma rigorosa a sintaxe proposta no enunciado do trabalho**. Espaços em branco, caracteres em posições inválidas e quebras de linha desnecessárias invalidam o arquivo de entrada. Adicionalmente, **não é permitido que os terminais ou as variáveis possuam um nome constituído por mais de um caractere**. Por exemplo, não podem existir variáveis denominadas “q0”. São permitidas no máximo **12 variáveis** no arquivo de entrada (para os terminais, não há limite).

Abaixo, segue um exemplo de arquivo de entrada **totalmente válido**. Nota-se que a sintaxe do enunciado do trabalho foi obedecida.

```
1 MinhaGLUD= ({A,B,C},{a,b,c},prod,A)
2 prod
3 A -> aB
4 B -> bC
5 A -> b
6 C ->
7 C -> cA
```

Abaixo, vemos três exemplos de arquivos de entrada **inválidos**. No primeiro exemplo, existe um espaço em branco desnecessário na linha 7. No segundo exemplo, há uma quebra de linha desnecessária, gerando uma linha vazia. No terceiro exemplo, percebe-se ausência de informação na primeira linha do arquivo (no caso, a variável inicial não foi indicada).

```
1 MinhaGLUD= ({A,B,C},{a,b,c},prod,A)
2 prod
3 A -> aB
4 B -> bC
5 A -> b
6 C ->
7 C -> cA
```

```
1 MinhaGLUD= ({A,B,C},{a,b,c},prod,A)
2 prod
3 A -> aB
4 B -> bC
5 A -> b
6 C ->
7 C -> cA
8
```

```
1 MinhaGLUD= ({A,B,C},{a,b,c},prod)
2 prod
3 A -> aB
4 B -> bC
5 A -> b
6 C ->
7 C -> cA
```

Abaixo, seguem outros exemplos de arquivos de entrada inválidos:

```
1 MinhaGlud= ({A,B,C},{a,b,c},prod,A)
2 prod
3 A ->aB
4 B -> bC
5 A -> b
6 C ->
7 C -> cA
```

Na linha 3, há um erro de sintaxe, pois não há espaço entre ">" e "a"

```
1 MinhaGlud= ({A,B,C},{a,b,c},prod,A)
2 prod
3 A -> aB
4 B -> bC
5 A -> b
6
7 C ->
8 C -> cA
```

Quebra de linha desnecessária na linha 6

```
1 MinhaGlud= ({A,B,C},a,b,c},prod,A)
2 prod
3 A -> aB
4 B -> bC
5 A -> b
6 C ->
7 C -> cA
```

Ausência de chaves na primeira linha

Especificamente para indicar produções do tipo $A \rightarrow \varepsilon$, o usuário deve realizar a quebra de linha (*linefeed*) imediatamente após o caractere ">", ou seja, não é permitido qualquer texto após o caractere ">". Caso a produção do tipo $A \rightarrow \varepsilon$ seja a última do arquivo (ou seja, presente na última linha), não deve existir nenhum caractere após o caractere ">" (também não deve haver quebra de linha), conforme mostrado abaixo.

```
1 MinhaGLUD= ({A,B,C},{a,b,c},prod,A)
2 prod
3 A -> aB
4 B -> bC
5 A -> b
6 C -> cA
7 C ->
```

Abaixo, vemos dois exemplos de arquivos **inválidos**. Na primeira imagem, houve inserção de um espaço em branco após o caractere ">". Na segunda, há presença de quebra de linha após a sétima linha, fazendo com que a linha 8 seja vazia e inválida (o que torna todo o arquivo de entrada inválido).

```
1 MinhaGLUD=({A,B,C},{a,b,c},prod,A)
2 prod
3 A -> aB
4 B -> bC
5 A -> b
6 C -> cA
7 C -> |
```

```
1 MinhaGLUD=({A,B,C},{a,b,c},prod,A)
2 prod
3 A -> aB
4 B -> bC
5 A -> b
6 C -> cA
7 C ->
8 |
```

Outros exemplos de erros **que são detectados pelo programa e que invalidam o arquivo de entrada** incluem:

- Ausência de qualquer informação essencial para a definição da GLUD na linha 1;
- Nomes de variáveis repetidos no conjunto das variáveis;
- Nomes de terminais repetidos no conjunto dos terminais;
- Nomes de variáveis incluídos no conjunto dos terminais ou vice-versa;
- Casos em que o usuário escolhe um nome de variável inicial que não está presente no conjunto de variáveis;
- Casos em que o usuário insere, em qualquer regra de produção um terminal ou uma variável que não foi definida nos conjuntos da primeira linha;
- Nome das regras de produção indicado na primeira linha distinto do nome indicado na segunda linha (atentar para espaços em branco desnecessários na segunda linha);
- Produções do tipo $A \rightarrow B$, em que A e B são variáveis;
- Ausência do sinal "=" após o nome da GLUD, na primeira linha.
- Presença de mais de dois caracteres do lado direito da produção (por exemplo, $A \rightarrow aBB$, $S \rightarrow aa$, etc).

Casos do arquivo de entrada **que são tolerados pelo programa incluem:**

- Repetir uma mesma produção na lista das regras de produções, a partir da terceira linha;
- Declarar uma variável ou um terminal nos conjuntos da primeira linha, mas não referenciar essa variável ou esse terminal em nenhuma das regras de produção, a partir da terceira linha.

Caso qualquer erro mencionado seja detectado pelo programa, ele é encerrado e a seguinte mensagem é exibida ao usuário:

```
Erro de sintaxe ou semantica no arquivo de entrada. Fim do programa.
```

2 - ARQUIVO DE ENTRADA CONTENDO A LISTA DE PALAVRAS A SEREM ANALISADAS

Qualquer palavra inserida em qualquer linha do arquivo de entrada com a lista de palavras será analisada (ou seja, o programa irá decidir se a palavra da linha pode ser gerada pela GLUD inserida pelo usuário no primeiro arquivo). Caso o usuário deseje testar a palavra vazia, deve ser feita uma quebra de linha de forma imediata na linha (deixando a linha totalmente vazia), sem qualquer espaço em branco. De forma similar, caso a palavra vazia seja a última do arquivo, não deve existir nenhum espaço em branco ou outro caractere na linha.

```
1 aaabbb
2 ababab
3
4 abbab
```

Exemplo de arquivo de entrada. Na terceira linha, foi indicada a palavra vazia.

```
1 aaabbb
2 ababab
3 abbab
4
```

Nesse outro exemplo, a palavra vazia foi indicada na quarta linha

```
1 aaabbb
2 ababab
3 abbab
4
```

Nesse outro exemplo, na quarta linha a palavra indicada NÃO é vazia, pois há um espaço em branco desnecessário

3 - EXECUÇÃO DO PROGRAMA

Após abrir o programa, o usuário deve, no menu inicial, inserir o nome do arquivo texto contendo a GLUD, além do arquivo texto contendo a lista de palavras a serem analisadas, conforme mostra a foto abaixo:

```
LINGUAGENS FORMAIS E AUTOMATOS

TRABALHO FINAL PARTE 2

Andrei Pochmann Koenich
Gustavo Spellmeier Neves
Pedro Company Beck

-----

Digite o nome do arquivo de entrada contendo a GLUD:
glud.txt

Digite o nome do arquivo de entrada contendo a lista de palavras:
palavras.txt_
```

Em seguida, o programa irá indicar na tela quais das palavras do segundo arquivo podem ser geradas pela GLUD do primeiro arquivo. O programa também irá indicar se a GLUD inserida gera uma linguagem vazia, finita ou infinita. Depois disso, o usuário pode optar por pressionar ESC e encerrar a execução do programa, ou pressionar qualquer outra tecla para voltar ao menu inicial e realizar uma nova operação, conforme mostrado abaixo.

```
PALAVRAS DO ARQUIVO DE ENTRADA QUE PODEM SER GERADAS PELA GLUD:

ab
abcb
abcb
abcbabc
abcbabcabcb
<palavra vazia>

-----

A GLUD gera uma linguagem regular INFINITA.

Pressione ESC para encerrar a execucao do programa.
Pressione qualquer outra tecla para realizar uma nova leitura._
```

Caso o usuário indique um arquivo de entrada inválido, o programa exibe uma mensagem de erro informando qual dos dois arquivos é inválido e encerra sua execução, conforme mostram as fotos abaixo.

```
Erro na abertura do arquivo texto contendo a GLUD. Fim do programa.
_
```

```
Erro na abertura do arquivo texto contendo a lista de palavras. Fim do programa.
_
```

Caso nenhuma palavra do arquivo de entrada escolhido pelo usuário possa ser gerada pela GLUD escolhida, o programa exibe a mensagem mostrada na imagem abaixo.

```
PALAVRAS DO ARQUIVO DE ENTRADA QUE PODEM SER GERADAS PELA GLUD:  
Nenhuma palavra do arquivo de entrada pode ser gerada pela GLUD inserida.  
-----
```