

Substitutiva da Segunda Prova de Construção de Compiladores.

Primeiro Semestre de 2001. Dept. de Computação – UFSCar.

José de Oliveira Guimarães.

Turma B (Quinta).

Lembre-se: justifique tudo a menos de menção em contrário.

Faça uma e apenas uma das duas questões abaixo.

1. (2.0) Explique o que é sistema de tempo de execução. Cite exemplos.

2. (2.0) Faça uma gramática com operadores binários $+$ e $*$ e operador únario prefixado $\&$ tal que:

- $+$ seja associado à esquerda;
- $*$ seja associado à direita;
- $\&$ tenha menor precedência do que $*$
- $*$ tenha menor precedência do que $+$.

Não é necessário justificar.

3. (2.0) Descubra se a gramática

$$E ::= F + E \mid F * F \mid F$$
$$F ::= N \mid E$$

é ambígua ou não. Justifique a sua resposta.

4. (2.0)

a) Retire a recursão à esquerda da seguinte gramática:

$$E ::= E \mid T \mid T$$
$$T ::= T \&\& F \mid T \wedge F \mid F$$
$$F ::= 0 \mid 1$$

Não é necessário justificar, mas deixe claros os passos que você seguir.

b) Fatore a gramática do item a).

5. (2.0) Faça

a) a expressão regular do JLex que reconhece uma palavra que comece por uma letra maiúscula e termine por um dígito entre 2 e 5. Pode haver qualquer letra entre a primeiro e o último caráter da palavra, exceto “a”. Como exemplo, as palavras “Atudobem2” e “Metodo4” deveriam ser reconhecidas. Já “ZeroGrau0” deve causar um erro, pois “a” aparece no meio da sentença e a palavra termina por 0.

b) o autômato finito que reconhece a palavra do item a) (apenas o desenho, não o código).

6. (2.0) Pode ser provado que se uma gramática G é $LL(1)$ então se $A ::= \alpha \mid \beta$ forem duas produções distintas de G e $\beta \xrightarrow{*} \epsilon$, então α não derivará qualquer forma sentencial começando com um terminal que pertence a $\text{follow}(A)$.

Explique porque esta regra é necessária para que G não seja ambígua. Para auxiliá-lo na resposta, encontre uma sentença que possua duas derivações à esquerda diferentes se esta regra for violada.