

Segunda Prova de Construção de Compiladores.
Primeiro Semestre de 2002.
Departamento de Computação – UFSCar.
José de Oliveira Guimarães.
Turma A (Terça).

Lembre-se: justifique tudo a menos de menção em contrário.
Entregue apenas a folha de respostas. Isto é, não entregue esta folha ou o rascunho.

1. (2.5) Explique porque a regra
Se $A ::= \alpha | \beta$, então ou $b \in \text{first}(\alpha)$ ou $b \in \text{first}(\beta)$
é necessária para que possamos construir um analisador recursivo descendente para uma gramática.
Isto é, se a regra não é obedecida, a gramática é ambígua.

2. (2.5) Desenhe a MEF e faça o código em Java que analise a seguinte expressão regular:
 $[0-9]^+(E|e)[a-z]^*$

3. (2.5) Fatore à esquerda e retire a recursão à esquerda da seguinte gramática:

```
S ::= RA | RB | c
A ::= Aa | Ab | c | d
B ::= b | c
R ::= r | s
```

4. (3.0) Calcule first e follow para todos os não terminais da gramática

```
P ::= S | S ; S
S ::= id = E
S ::= print ( L )
E ::= E + E | N | id
L ::= E | E , L
```

$\text{first}(X)$ é computado como:

1. se X for um terminal, $\text{first}(X) = \{ X \}$;
2. se $X ::= \epsilon$ for uma produção, adicione ϵ a $\text{first}(X)$;
3. se X for não terminal e $X ::= Y_1 Y_2 \dots Y_k$ for uma produção, então coloque f em $\text{first}(X)$ se, para algum i , $f \in \text{first}(Y_i)$ e $\epsilon \in \text{first}(Y_j)$, $j = 1, 2, \dots, i - 1$. Se $\epsilon \in \text{first}(Y_j)$, $j = 1, 2, \dots, k$, então coloque ϵ em $\text{first}(X)$.

$\text{follow}(X)$ é computado como:

1. coloque eof em $\text{follow}(S)$, onde S é o símbolo inicial;
2. se houver uma produção da forma $A ::= \alpha X \beta$, então $\text{first}(\beta)$, exceto ϵ , é colocado em $\text{follow}(X)$;
3. se houver uma produção da forma $A ::= \alpha X$ ou uma produção $A ::= \alpha X \beta$ e $\epsilon \in \text{first}(\beta)$, então colocaremos os elementos de $\text{follow}(A)$ em $\text{follow}(X)$.