

Curso:

Prova de Compilação

Data: 21 de Julho de 2009

Nome:

Nº de Estudante: B. I. nº

Assinatura do Vigilante:



LEIA ATENTAMENTE as instruções para a resolução do exame:

- O exame é constituído por quatro questões, com as seguintes cotações:
 1. 6 valores
 2. 6 valores
 3. 5 valores
 4. 3 valores
- O exame é **SEM CONSULTA**. Todos os elementos necessários à resolução são fornecidos no enunciado.
- Utilize esferográfica azul ou preta para responder às questões. Respostas a lápis não serão consideradas.
- Leia o enunciado de todas as questões antes de começar a responder.
- O tempo de resolução do exame é de duas horas e trinta minutos.
- O exame é constituído por 14 páginas e termina com a palavra **FIM**.
- Se o seu exemplar não estiver completo ou nele se verificar qualquer outra deficiência, por favor dirija-se ao professor vigilante.
- A correcção do exame terá em conta critérios de proficiência e legibilidade da caligrafia.

RESERVADO PARA A *Universidade Aberta*

Data: 21 de Julho de 2009

Prova de **Compilação**

Classificação: ()

Prof. que classificou a prova:

1. Considere o alfabeto $V = \{a, b, c\}$ e o conjunto das frases que terminam na sequência “bab”.

- Escreva a respectiva expressão regular.
- Escreva o respectivo autómato finito não determinista.
- Converta o autómato anterior num autómato finito determinista (AFD), apresentando todos os passos intermédios.

2. Considerando a seguinte gramática G:

$A \rightarrow B \text{ or } A \mid B$

$B \rightarrow C \text{ and } B \mid C$

$C \rightarrow (A) \mid \text{true} \mid \text{false}$

- Construa as tabelas de ações e de saltos do analisador sintático ascendente LR, pelo método LALR.
- Usando estas tabelas, mostre a sequência de reconhecimento da frase **true or false and false .**

3. Considerando a implementação de um compilador de uma nova linguagem chamada O-TEMPO.

- Implemente um analisador léxico que reconheça os seguintes *tokens*:

<i>Token</i>	Descrição
NEVE	Sequências de letras minúsculas de comprimento não superior a 4
SOL	Sequências de 3 letras maiúsculas iniciadas pela letra S
CINZENTO	Todos os prefixos da palavra cinzento
NEVOEIRO	Qualquer sequência onde não apareçam dois e 's consecutivos
GRANIZO	Todas as sequências de letras minúsculas que terminem em zo
GELO	Qualquer sequência com letras da palavra gelo
CHUVA	Qualquer sequência de letras onde ch é prefixo e v é um factor
NUVEM	Todas as sub-sequências (<i>substrings</i>) que se obtêm de nuvem tirando de zero a cinco letras

- Reescreva a gramática seguinte, que representa expressões escritas na linguagem O-TEMPO:

E → **E** : **E** | **E** ^ **E** | **!** **E** | (**E**) | **TOKEN**
TOKEN → **NEVE** | **SOL** | **CINZENTO** | **NEVOEIRO** | **GRANIZO** |
GELO | **CHUVA** | **NUVEM**

onde **a**, **b** e **o** são operadores, tendo em consideração que:

- O operador **!** tem precedência sobre **^** (associativo à esquerda), e este sobre **:** (associativo à direita).
- A nova gramática não deverá ser ambígua nem conter recursividade à esquerda.

4. Optimize o seguinte código gerado em TAC (*Three Address Code*), explicitando o tipo de otimização que está a fazer:

```
b = 7 - 3
t1 = b * 2
t2 = a + t1
t3 = t2 * b
t4 = t3 / 2
t5 = t4 + c
t6 = t2 * b
d = t6 + t4
```


FIM