UNIVERSIDADE ABERTA
www.uab.pt

Ministério da Ciência, Tecnologia e Ensino Superior

U.C. 21018

Compilação

13 de Setembro de 2012

-- INSTRUÇÕES --

- O estudante deverá responder à prova na folha de ponto e preencher o cabeçalho e todos os espaços reservados à sua identificação, com letra legível.
- No fim da prova, poderá ficar na posse do enunciado.
- Verifique no momento da entrega da(s) folha(s) de ponto se todas as páginas estão rubricadas pelo vigilante. Caso necessite de mais do que uma folha de ponto, deverá numerá-las no canto superior direito.
- Em hipótese alguma serão aceites folhas de ponto dobradas ou danificadas.
- Exclui-se, para efeitos de classificação, toda e qualquer resposta apresentada em folhas de rascunho.
- Os telemóveis deverão ser desligados durante toda a prova e os objectos pessoais deixados em local próprio da sala de exame.
- Utilize unicamente tinta azul ou preta.
- A prova é constituída por **3** páginas (esta página de rosto e duas com as questões), contém 5 questões, sem consulta, e termina com a palavra **FIM**. Verifique o seu exemplar e, caso encontre alguma anomalia, dirija-se ao professor vigilante nos primeiros 15 minutos da mesma, pois qualquer reclamação sobre defeito(s) de formatação e/ou de impressão que dificultem a leitura não será aceite depois deste período.

Duração: 150 minutos

1ª Questão (4 valores)

Considere a seguinte gramática:

```
P -> Var Statement
Var -> type id .
Statement -> id <- Expr .
Expr -> Moon & Expr
Moon -> Sun # Moon
Sun -> ( Expr ) | id
```

Construa as tabelas de acções e saltos do analisador sintáctico ascendente LR, pelo método SLR.

2ª Questão (4 valores)

Apresente a especificação *flex* para um analisador léxico que identifique os seguintes tokens e imprima no ecrã o respectivo token ID. No caso dos identificadores, o programa deve imprimir também, entre parênteses, o nome do identificador.

- Identificador (token ID: VAR) – uma letra seguida de letras e dígitos
- Constante lógica (token ID: CLOG) – **true** ou **false**
- Constante numérica em notação hexadecimal (token ID: CHEX) – uma sequência em notação hexadecimal (podendo as letras ser maiúsculas ou minúsculas).

3ª Questão (4 valores)

Apresente o código intermédio, em notação TAC (*three address code*) correspondente ao seguinte excerto de código em linguagem C:

```
int a[10];
int i = 1;

a[0] = 0;
while (i < 10){
  a[i] = i + a[i-1];
  i++;
}
```

4ª Questão (4 valores)

Apresente a especificação *bison* para um analisador sintáctico de expressões sobre strings, que devolva o resultado da avaliação da expressão analisada. As constantes (numéricas e strings) são identificadas pelo analisador léxico, sendo que o analisador sintático deverá considerar as seguintes operações:

- Concatenação de strings: o resultado de *s1 + s2* é uma string com a sequência de caracteres de *s1* seguida da sequência de caracteres de *s2*. **Ex.** "abc"+"123" é "abc123"
- Repetição: o resultado de *n* * *s*, em que *n* é um número inteiro e *s* é uma string, e é a concatenação de *s* com ela própria *n* vezes. Ex. 3*"ab" é "ababab"

Considere ainda que o operador * tem precedência sobre + e que pode usar parêntesis para alterar a ordem de avaliação

5ª Questão (4 valores)

Optimize o seguinte código gerado em TAC (*three address code*), explicitando o tipo de optimização que está a fazer:

```
a = 2 * 2
t1 = a - 2
t2 = t1 + 2
t3 = t1 * a
t4 = t2 + t3
t5 = t4 + b
t6 = t5 + t4
c = t5 * 2
```

FIM