

Processamento de Linguagens – MiEI

Exame de Recurso

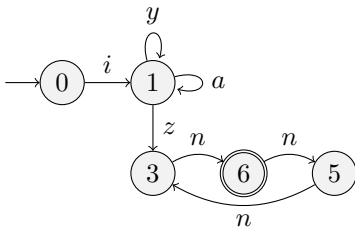
18 de Junho de 2018 (13h00)

Dispõe de **2:00 horas** para realizar este teste.

Questão 1: Expressões Regulares e Autómatos (3v)

Responda às seguintes alíneas:

a) Qual a expressão regular correspondente ao seguinte autómato:



b) Para ajudar a decorar o número da porta da namorada (nº 958), optou por procurar n'Os Lusíadas sequências de 3 palavras com comprimentos 9, 5, 8 obtendo: "valerosas obras exercita", "Ocidental praia Lusitana" (o que, para além de facilitar fixar o número, funciona também de facilitador de conversa).

Indique como é que usando **grep**, conseguiu fazer tal façanha! (Evite soluções que devolvam palavras parciais).

c) Considere as seguintes expressões regulares abaixo, escritas em notação do Flex:

- 1 (dia|DIA)
- 2 [dia|DIA]
- 3 [Dd][Ii][Aa]
- 4 (?i:dia)
- 5 dia/DIA
- 6 dia{1,3}
- 7 {dia}

Apresente todas as frases que cada uma delas *apanha*. No caso de serem muitas, apresente 6 das mais curtas. Comente quando necessário

Questão 2: Filtros de Texto em Flex e GAWK (4v = 2+2)

Especifique filtros de texto com base em expressões regulares e regras de produção (padrão-ação) para resolver as seguintes alíneas:

a) *Matrículas de outro mundo*

Num país alheio, as matrículas dos automóveis seguem os seguintes requisitos:

- Uma matrícula tem 8 dígitos;
- Os 8 dígitos estão divididos em 4 partes iguais de 2 dígitos por um separador que pode ser '.', '-' ou ':';
- Cada matrícula só pode ter uma espécie de separador;
- Os separadores têm de ter dígitos antes e depois, não há espaços.

Especifique um filtro em flex que apanhe e contabilize as matrículas num texto.

O texto contém ainda vários comentários multilinha em sintaxe C (`/*...*/`) que deve ignorar.

b) Para pôr ordem na casa construiu-se uma listagem dos livros com o seguinte aspecto:

```
Volfrâmio : Aquilino : comprado 2009
Os Maias : Eça : presente da avó Maria
Os Lusíadas : Camões : feira do livro 2011
Os Maias: Eça de Queirós : versão anotada
```

Constatou-se que há algumas repetições que pretende encontrar.

Escreva um filtro usando o `gawk` para anote títulos de livros duplicados de modo a obter:

```
Volfrâmio : Aquilino : comprado 2009
Os Maias : Eça : presente da avó Maria
Os Lusíadas : Camões : feira do livro 2011
==2==Os Maias: Eça de Queirós : versão anotada
```

Notas:

- (1) Nas linha contendo um título repetido, anotou-se o número da primeira ocorrência, de acordo com o exemplo.
- (2) Não analise nem altere linhas com menos de 5 caracteres.

Questão 3: Desenho/especificação de uma Linguagem (4v=2+1+1)

Pretende-se uma linguagem para descrever uma Sociedade Anónima e os seus elementos caracterizadores ou constituintes, de acordo com a seguinte definição:

Uma Sociedade Anónima (SA) tem um nome, uma localização e um ramo de negócio (agricultura, indústria ou comércio); uma SA é uma empresa que é detida por um mais sócios que possuem o capital dividido em cotas. Esses sócios reúnem-se em assembleia geral (AG) de tempos a tempos, sendo essa AG presidida pelo sócio que é eleito entre os sócios maioritários (aqueles que possuem a maior cota). Toda a SA existe com um fim: realizar um determinado negócio (o seu objetivo). O Administrador é quem define as estratégias de negócio. Porém a empresa é gerida no seu dia a dia pelo Gerente que é um dos funcionários da empresa. Além do Gerente, os outros funcionários ou trabalham na Produção (realizando o produto que é objeto do negócio), ou no Escritório (realizando as tarefas apoio à Gestão), ou no Armazém (realizando as tarefas apoio à Produção).

- a) Desenhe uma GIC a seu gosto para definir a linguagem pretendida.
- b) Usando a gramática especificada em cima apresente uma frase exemplo e construa a respectiva Árvore de Derivação.
- c) Usando a notação do Flex especifique um analisador léxico para esta linguagem.

Questão 4: Gramáticas e Parsing (7v)

Considere a gramática independente de contexto, G , abaixo apresentada, ou melhor um seu fragmento, que permite gerir o movimento de partidas e chegadas de aviões num aeroporto. Note ainda que os símbolos terminais T (note que ϵ representa a sequência vazia epsilon) e não-terminais NT estão definidos antes do conjunto de produções P , sendo L o seu axioma (ou símbolo inicial).

```

T = { '|', '-', ':', PARTIDAS, CHEGADAS, data, id, str, hora }
NT = { Trafego, Partidas, Chegadas, Avioes, Aviao, NumVoo, Companh, Origem }
P = {
p1: Trafego -> data Partidas Chegadas
p2: Partidas -> PARTIDAS Aviao Avioes
p3: Avioes -> '|' Aviao Avioes
p4:         | &
p5: Chegadas -> CHEGADAS .....
p6: Aviao -> NumVoo '-' Companh '-' Origem ':' hora
p7: NumVoo -> id
p8: Companh -> str
p9: Origem -> str
}

```

Neste contexto e após analisar a G dada, responda às alíneas seguintes.

- Após estender a G dada, construa o respetivo **autómato LR(0)** e identifique todas as **situações de conflito** que eventualmente ocorram.
- No âmbito do Parsing LR(0), ao ler o símbolo terminal **CHEGADAS** qual vai ser a ação de redução indicada, isto é, qual a produção a reduzir, quantos estados recua e qual o símbolo NT que determina o GOTO.
- Escreva as funções de um parser RD-puro (recursivo-descendente) para reconhecer os 3 símbolos não-terminais **Partidas** e **Avioes**.
- Calcule o Lookahead das produções p1 a p4 e escreva a tabela de decisão LL(1) só para os NTs nelas envolvidos.
- Acrescente as necessárias Ações Semânticas a G para gerar uma pagina HTML com as Partidas de voos; junte ainda à pagina o numero de voos registados.

Questão 5: Geração de Código (2v)

Recorde-se que nas aulas práticas foram definidas 5 variantes de **Instrução** para a linguagem imperativa *C-like*

```

inst: inst-atrib           { $$ = $1; }
    | inst-escrever       { $$ = $1; }
    | inst-ler            { $$ = $1; }
    | inst-condicional    { $$ = $1; }
    | inst-ciclo          { $$ = $1; }
    ;

```

Imagine então que agora se quer acrescentar uma nova instrução iterativa do tipo *for(var = int to int) { ... }*

```

inst-ciclo:
.....
    | inst-for             { $$ = $1; }
    ;
inst-for: "for" '(' Id '=' Num "to" Num ')' '{' insts '}' { ASfor }
    ;

```

Escreva então a ação semântica **ASfor** para gerar o código Assembly da VM que implementa este novo ciclo.