



## Compiladores: Prova 2

Nome:

Matrícula:

Data:

Observações:

- (a) A prova é individual e sem consulta, sendo vedado o uso de calculadoras e de telefones celulares.
- (b) A interpretação dos comandos das questões faz parte da avaliação.
- (c) A nota da prova será igual a 0 (zero) caso o estudante consulte algum material durante a prova, ou receba ou ofereça qualquer ajuda a outro estudante durante a prova.
- (d) O gabarito deve ser preenchido com caneta esferográfica azul ou preta. Em uma questão de múltiplas escolhas, múltiplas marcações anularão a referida questão.
- (e) O grampo da prova não deve ser removido. Caso o grampo seja removido, a nota da prova será igual a 0 (zero).

### Parte A. (70 pontos) Assinale a alternativa correta.

#### 1. Considere as afirmações abaixo.

- I. Um token é um símbolo terminal de uma gramática.
- II. Um mesmo token pode ser representado por lexemas distintos.

Podemos afirmar que

- (X) ambas afirmações estão corretas
- (B) ambas afirmações estão incorretas
- (C) a afirmativa I está correta e a afirmativa II está incorreta
- (D) a afirmativa I está incorreta e a afirmativa II está correta

#### 2. Considere o seguinte trecho de código, escrito em linguagem C:

```
int _ = 1;
```

Pode-se afirmar que este trecho

- (A) contém um erro léxico
- (B) contém um erro sintático
- (C) contém um erro semântico
- (X) não contém erros

#### 3. Considere três estratégias de tratamento da buferização:

- I. usar um gerador léxico, que será o responsável pela buferização.
- II. utilizar uma linguagem de programação e a buferização a ela atrelada.
- III. implementar a buferização explicitamente em linguagem de montagem.

Qual sequência de identificadores elenca estas três estratégias, da mais complexa para a mais simples?

- (A) I, II, III
- (B) I, III, II
- (C) II, I, III
- (X) III, II, I

#### 4. Considere, no contexto da implementação da buferização, o uso de um par de buffers, com $N$ caracteres cada. Assuma que cada caractere ocupe 1 byte de memória. Qual é a área ocupada por este par de buffers, se $N = 8$ e forem usados caracteres sentinelas para sinalizar o final de cada bloco?

- (A) 12
- (B) 14
- (C) 16
- (X) 18

5. Considere os conjuntos

$$A = \{\heartsuit, \diamondsuit, \spadesuit, \clubsuit\}$$

e

$$B = \{n \in \mathbb{N} \mid n \text{ divide } 2^{100}\}$$

É verdade que

- (X) ambos conjuntos são alfabetos
  - (B) apenas o conjunto  $A$  é um alfabeto
  - (C) apenas o conjunto  $B$  é um alfabeto
  - (D) nenhum dos dois conjuntos é um alfabeto
6. Seja  $s$  uma cadeia sobre um alfabeto  $\Sigma$  e  $t$  a cadeia obtida a partir da remoção de zero ou mais caracteres do início de  $s$ . Então, em relação a  $s$ , a cadeia  $t$  é

- (A) prefixo
  - (X) sufixo
  - (C) subcadeia
  - (D) subsequência
7. Seja  $s = \text{"babbabbab"}$ . Qual é o menor natural  $n$  tal que existe uma cadeia  $t$  com  $s = t^n$ ?

- (X) 1
  - (B) 2
  - (C) 3
  - (D) 4
8. Sejam  $L$  e  $M$  duas linguagens. Qual é a notação para linguagem abaixo?

$$N = \{s \mid s \in L \vee s \in M\}$$

- (X)  $N = L \cup M$
  - (B)  $N = LM$
  - (C)  $N = L^*$
  - (D)  $N = L^+$
9. Seja  $L = \{a, b\}$  e  $M = \{x, y\}$ . A cadeia  $abax$  pertence a qual linguagem?

- (X)  $L^3M$
- (B)  $ML^3$
- (C)  $L^2M^2$
- (D)  $LM^2L$

10. Sejam  $r, s, t$  expressões regulares e  $\epsilon$  a cadeia vazia. A expressão

$$r(s \mid t) = (rs \mid rt)$$

corresponde a qual propriedade das expressões regulares?

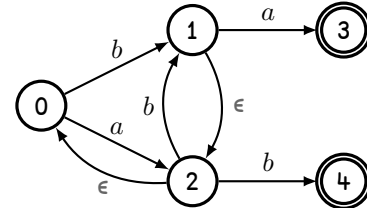
- (A) associativa
- (X) distributiva
- (C) idempotente
- (D) elemento neutro

11. Dadas quatro cadeias

$$s_1 = \epsilon, s_2 = baab, s_3 = bbab \text{ e } s_4 = bbbb,$$

quantas dentre estas cadeias fazem parte da linguagem definida pela expressão regular  $b^*(ab^?)^+$ ?

- (A) 0
  - (B) 1
  - (X) 2
  - (D) 3
12. Considere o grafo de transições do AFN abaixo.



No grafo de transições do AFD equivalente, obtido por meio do algoritmo de conversão de AFN para AFD apresentado em aula, quantas são os estados de aceitação?

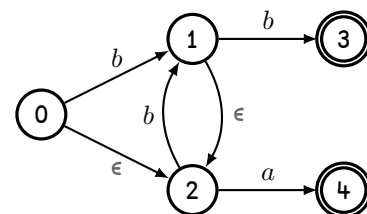
- (A) 1
  - (X) 2
  - (C) 3
  - (D) 4
13. Dada a gramática livre de contexto

$$S \rightarrow aSb \mid bTa$$

$$T \rightarrow aT \mid bS \mid \epsilon,$$

quantos são os passos da derivação, a partir de  $S$ , da cadeia  $aabaabb$ ?

- (A) 3
  - (B) 4
  - (X) 5
  - (D) 6
14. Considere o diagrama de transições do AFN abaixo:

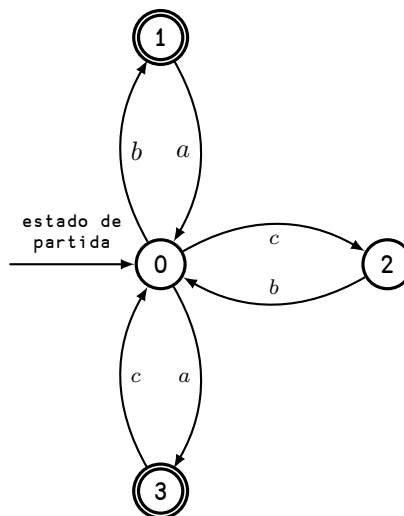


Seja  $X = \{1, 3\}$ . Qual é o Fechamento- $\epsilon(X)$ ?

- (A)  $\{4\}$
- (B)  $\{0, 2\}$
- (C)  $\{1, 2\}$
- (X)  $\{1, 2, 3\}$

Parte B. (30 pontos) Resolva a questão a seguir.

15. Considere o AFD abaixo:



Implemente, em C, C++ ou Python, a função `is_valid()`, que recebe uma string `s` como parâmetro e que retorna verdadeiro, se todos os caracteres de `s` compõem uma cadeia válida da linguagem definida pelo AFD; ou falso, caso contrário. Use apenas elementos básicos da linguagem, sem recorrer a bibliotecas externas ou expressões regulares. Use o diagrama como guia: a função deve aceitar todas as cadeias que o diagrama aceita, e recusar todas as cadeias que o diagrama recusa.

Seguem abaixo alguns testes unitários para esta função.

```
assert(is_valid("bacbaca"));
assert(is_valid("bab"));
assert(is_valid("cba"));
assert(not is_valid("cbacba"));
assert(not is_valid(""));
```

**Importante:** Escreva o código com letra legível, de forma organizada e clara, usando as folhas pautadas do final da prova. O código não deve exceder 50 linhas.

```
1
2 int in_valid(const char *s)
3 {
4     int state = 0;
5     char *p = NULL;
6
7     for (p = s; *p; ++p)
8     {
9         if (*p != 'a' && *p != 'b' && *p != 'c')
10             return 0;
11
12         switch (state) {
13             case 0:
14                 if (*p == 'a')
15                     state = 3;
16                 else if (*p == 'b')
17                     state = 1;
18                 else
19                     state = 2;
20                 break;
21
22             case 1:
23                 if (*p == 'a')
24                     state = 0;
```

```

25         else
26             return 0;
27
28     case 2:
29         if (*p == 'b')
30             state = 0;
31         else
32             return 0;
33
34     case 3:
35         if (*p == 'c')
36             state = 0;
37         else
38             return 0;
39     }
40 }
41
42 return state == 1 || state == 3;
43 }

```

```

1 def is_valid(s):
2
3     if len(s) % 2 == 0 or s[-1] != 'a' or s[-1] != 'b':
4         return False
5
6     for i in range(0, len(s) // 2):
7         t = s[2*i:2*i + 2]
8
9         if t not in [ 'ba', 'cb', 'ac' ]:
10             return False
11
12 return True

```