Iniciado em	Tuesday, 10 Nov 2020, 10:10
Estado	Finalizada
Concluída em	Tuesday, 10 Nov 2020, 11:47
Tempo empregado	1 hora 37 minutos
Avaliar	**4,25** de um máximo de 10,00(**43**%)

Questão 1
Correto
Atingiu 0,20 de 0,20

Todo problema que pode ser representado como um problema de linguagem é um problema computável.

Escolha uma opção:
- Verdadeiro
- Falso ✓

A resposta correta é 'Falso'.

Questão 2
Correto
Atingiu 0,20 de 0,20

Uma máquina de Turing multifitas pode ser simulada por uma máquina de Turing de fita única.

Escolha uma opção:
- Verdadeiro ✓
- Falso

A resposta correta é 'Verdadeiro'.

Questão 3
Incorreto
Atingiu 0,00 de 0,20

A possibilidade de transformar Máquinas de Turing em "código" que pode ser executado por outra Máquina de Turing demonstra que o conjunto de Máquinas de Turing é não enumerável.

Escolha uma opção:
- Verdadeiro ✗
- Falso

A resposta correta é 'Falso'.

Questão 4

Incorreto

Atingiu 0,00 de 0,20

Ao contrário das Linguagens Recursivamente Enumeráveis, nas Linguagens Recursivas o número de sentenças de tamanho n é sempre finito.

Escolha uma opção:

- Verdadeiro ✗
- Falso

A resposta correta é 'Falso'.

Questão 5

Incorreto

Atingiu 0,00 de 0,20

Linguagens Turing Reconhecíveis e Linguagens Computáveis são a mesma classe de linguagens.

Escolha uma opção:

- Verdadeiro
- Falso ✗

A resposta correta é 'Verdadeiro'.

Questão 6

Incorreto

Atingiu 0,00 de 0,20

Problemas não computáveis podem ser correlacionados com Linguagens Recursivamente Enumeráveis.

Escolha uma opção:

- Verdadeiro ✗
- Falso

A resposta correta é 'Falso'.

Questão 7

Correto

Atingiu 0,20 de 0,20

Uma Máquina de Turing Não Determinística reconhece a pertinência de uma palavra à uma linguagem se todos os ramos de computação findam em um estado de aceitação.

Escolha uma opção:

- Verdadeiro
- Falso ✓

A resposta correta é 'Falso'.

Questão 8

Correto

Atingiu 0,20 de 0,20

Uma Máquina de Turing Não Determinística verifica a não pertinência de uma palavra à uma linguagem se todos os ramos de computação findam em um estado de rejeição.

Escolha uma opção:

- Verdadeiro ✓
- Falso

A resposta correta é 'Verdadeiro'.

Questão 9

Correto

Atingiu 0,20 de 0,20

Assim como AFD e AFND, Autômatos de Pilha Determinísticos (APD) e Autômatos de Pilha Não Determinísticos (APND) são equivalentes.

Escolha uma opção:

- Verdadeiro
- Falso ✓

A resposta correta é 'Falso'.

Questão 10

Incorreto

Atingiu 0,00 de 0,20

Se uma linguagem L é Turing Reconhecível e seu complemento $\overline{L}$ é Co-Turing Reconhecível então a linguagem L pode ser decidida por uma MT.

Escolha uma opção:

- (selecionado) Verdadeiro ✗
- Falso

A resposta correta é 'Falso'.

Questão 11

Completo

Atingiu 0,50 de 1,00

Apresente uma definição formal de um Autômato Linearmente Limitado.

Um Autômato Linearmente Limitado pode ser definido por uma octupla (SIGMA, S, T, S0, F, V, <, >), onde:

SIGMA = conjunto de símbolos de entrada (alfabeto);
S = conjunto finito de estados;
T = função de transição, tal que T: S x (alfabeto completo) → 2S × (alfabeto completo) x {<- , ->}, onde "alfabeto completo" é a união do alfabeto de entrada, alfabeto auxiliar e simbolos de inicio e fim de fita;
S0 é o estado inicial, tal que S0 pertence a S
F é o conjunto de estados finais, tal que F esta contido em S
V é o alfabeto auxiliar, que pode ser vazio ou não;
< e > são os simbolos marcadores de inicio e fim da fita, respectivamente

Comentário:
que tal me apresentar a tua própria definição. Essa da wiki é boa, mas queria uma vinda de você!

Questão 12

Completo

Atingiu 0,50 de 1,00

Apresente uma definição formal de um Enumerador. Considere-o como um tipo de MT de duas fitas que usa a sua segunda fita como impressora. Inclua uma definição da linguagem enumerada.

Um enumerador nada mais é que uma MT com duas fitas: uma de entrada e uma de impressão. Logo, podemos definir um enumerador como a octupla (Q, SIGMA, GAMMA1, GAMMA2, Q0, U, F, T)

Q = conjunto finito de estados;
SIGMA = conjunto de simbolos;
GAMMA1 = alfabeto da fita 1 (entrada);
GAMMA2 = alfabeto da fita 2 (impressao);
Q0 = estado inicial, tal que Q0 pertence a Q;
U = simbolo de vazio (branco);
F = conjunto de estados finais, tal que F esta contido em Q;
T = função de transição, tal que T: Q x GAMMA -> Q x GAMMA x {<- , ->}, onde <- e -> são os movimentos do cabeçote para a esquerda e para a direita, respectivamente.

A linguagem enumerada vai ser a linguagem que a MT (enumerador) vai "imprimir" na fita. O enumerador vai "rodar" e, toda vez que houver uma saida, comparar essa com a entrada. Caso sejam iguais, imprimir essa saída na fita

Comentário:
Caso sejam iguais? então só vai imprimir a cadeia original... Para enumerar, a entrada é ignorada. roda E por i passos, imprime todas as cadeias com tamanho i ou menor e incrementa i e repete

Questão 13

Completo

Atingiu 0,50 de 1,00

É decidível se duas Linguagens Regulares L_1 e L_2 são complementares? Se sim, cite todos os passos necessários para essa decisão (verificação); senão, justifique.

Sim.

Seja M decisor de L1, ou seja, para toda entrada em M, ela aceita ou rejeita a entrada, aceitando aquelas que pertencem à L1, sendo A o estado de aceitação e R o estado de rejeição.

Seja M' a máquina que vai decidir L2, com estados A e R assim como M.

Rodamos M e M' paralelamente. Se quando M parar em A, M' parar em R, e quando M parar em R, M' parar em A, elas são complementares.

Comentário:
Oi Felipe, há uma falácia na tua prova. O conjunto de possíveis entradas é infinito, ou seja, ao rodar M1 e M2 para uma mesma entrada, pode ser que M1 aceite e M2 rejeite, mas pode ser que, mesmo e apesar disso, as linguagens não sejam complementares. Para construir uma prova nesta linha, precisaria tomar M1 e M2, completá-los, determinizá-los, e depois minimizá-los e ai por inspeção, verificar se são iguais. Mas ainda não temos todo esse ferramental... Para essa construção, um caminho simples seria se apoiar em EQ_{afd} e na vacuidade (via diferença simétrica)

Questão 14

Completo

Atingiu 0,25 de 1,00

Prove que uma Linguagem é Decidível se e somente se alguma Máquina de Turing Não Determinística a decide.

Por definição, uma linguagem decidível é aquela que tem uma MT que a decide.

Sabemos também que é possível criar uma máquina de turing deterministica (D) equivalente à uma máquina de turing não deterministica (N), fazendo com que D tente todos os ramos da computação não-determinística de N.

Ou seja, dizer que uma Linguagem é Decidivel sse uma MT ND à decide é a mesma coisa dizer que uma linguagem é decidível se uma maquina de turing à decide, o que sabemos por definição que é verdade.

Comentário:
Há detalhes importantes não esclarecidos. Uma MTND ao ser simulada por uma MTD pode nunca findar a simulação (devido ao endereçamento na 3a fita. A premissa da decidibilidade é importante, pois se a linguagem é decidível, todo ramo finda e a execução pode acontecer em profundidade e não em largura.

Questão 15

Completo

Atingiu 1,00 de 1,00

Conceitue e exemplifique (usando exemplos relacionados com a Teoria das Ling. Formais) Problemas Decídíveis e Problemas Indecidíveis, justificando a decidibilidade/indecidibilidade dos problemas usados como exemplo.

Problema decidivel é um tipo de problema que tem como resposta sim ou não, ou em ling. formais, que retorna se uma sequencia de simbolos pertence ou não à uma linguagem. A partir daqui, podemos definir problemas indecidiveis, ou seja, aqueles pra qual não conseguimos obter tal resposta (pela impossibilidade de construir um algoritmo que possibilite tal resposta)

Exemplo de um problema decidivel:

"dado dois numeros, x e y, retornar se x é multiplo (por um numero inteiro positivo diferente de zero) de y". Criar uma MT que:

1. Compare x com y. Se x for maior que y, rejeite, se for igual, aceite. Se não, continue
2. Some x com x
3. volte ao passo 1.

Exemplo de problema não decidivel:

O problema da parada.

Seja Amt = {<M, w> | M é uma MT e M aceita w} , ou seja, dada uma descrição de uma MT e uma entrada, decidir se a MT vai terminar a computação ou vai computar infinitamente.

Seja U uma MT que recebe as duas entradas acima (uma descricao de uma MT e uma entrada w)

U = “Sobre a entrada <M, w>, onde M é uma MT e w é uma cadeia:
1. Simule M sobre a entrada w.
2. Se M em algum momento entra no seu estado de aceitação, aceite; se M em algum momento entra em seu estado de
rejeição, rejeite.”

É importante notar que, se M entra em loop com a entrada w, então U vai entrar em loop com a entrada <M, w>, por isso U não decide a Amt

Comentário:

Questão 16

Completo

Atingiu 0,00 de 1,00

Por que há linguagens não computáveis? Como é possível provar sua existência?

Porque existem funções de transição que são extremamente complexas, impedindo que seja computável devido à sua complexibilidade.
Uma maneira de provar a existência de linguagem não computável é utilizando o problema da parada, que, dado um algoritmo de entrada, retorna se esse algoritmo vai entrar em loop infinito ou não.

Comentário:
não computabilidade está relacionado a ausência de MT...

Questão 17

Completo

Atingiu 0,50 de 2,00

Sejam P e Q dois problemas quaisquer. Assumindo como premissas cada um dos itens abaixo, construa argumentos e/ou contra-exemplos para responder ao que se pede:

1) Premissa: P é uma Linguagem Regular e Q é uma Linguagem Livre de Contexto.

Responda: $P \cap Q$ é uma Linguagem Regular?

2) Premissa: P é uma Linguagem Recursiva.
Responda: $\overline{P}$ é Computável? $\overline{P}$ é Decidível?

3) Premissa: $P \leq Q$ (P é redutível por mapeamento a Q) e Q é uma Liguagem Recursiva.
Responda: P é uma Linguagem Recursiva?

4) Premissa: P é computável e $\overline{P} \leq P$ ($\overline{P}$ é redutível a P).
Responda: $\overline{P}$ é computável? $\overline{P}$ é Decidível?

1. Sim. O conjunto das linguagens regulares está contido no conjunto das linguagens livres de contexto. A interseção entre P e Q será o próprio P, que é uma linguagem regular

2. Sim e Sim. Uma linguagem recursiva é a mesma coisa que uma linguagem decidível, ou seja, uma linguagem que é decidida por uma MT. Se ela é decidivel, ela também é computável, pois uma MT a computa para decidi-la.

3. Sim, por definição de redução, se Q é uma linguagem derivada da redução de P, e Q é decidivel (linguagem recursiva), então P é decidível

4. Não e Não. Pela definição dada em aula de redução, sabemos que o caminho oposto da alternartiva 3 acima não é necessariamente váida.

Comentário:
1) seja a*b* \cap a^n b^n -> não regular.

2) O complemento é decidível?

4) Se uma linguagem é redutível a outra, então a solução pode ser usada para resolver a primeira. Se ambas forem turing reconhecíveis P e P complemento, é possível construir uma MT que decide P.