

[Meus Cursos](#) / [Meus cursos](#) / [PL-APNP-2021/2-Codificacao-e-Compressao-de-Dados-CSEE](#)

/ [Sem 6- Métodos de Dicionário, Lempel-Ziv e Derivados](#) / [Tarefa 5 - Métodos de Dicionário](#)

Iniciado em quarta, 13 abr 2022, 00:10

Estado Finalizada

Concluída em quinta, 14 abr 2022, 01:03

Tempo 1 dia

empregado

Avaliar Ainda não avaliado

Questão **1**

Correto

Atingiu 1,00 de 1,00

A frase 'sir_sid_eastman_easily_teases_sea_sick_seals' é codificada com LZ 77 usando-se um SB (Search Buffer) e um LAB (Look Ahead Buffer) de dezesseis elementos cada. Em aula foram apresentados os primeiros quatro passos do processo de codificação. Se a mensagem segue sendo codificada os códigos para os passos oito a doze seriam:

Passo 8:

(6,1,t

✓ Passo 9:

(0,0,m)

✓ Passo 10

(4,1,n)

✓ Passo 11

(8,4,i)

✓ Passo 12:

(0,0,l)

✓ . Coloque as respostas no formato (deslocamento,numerodeiguais,novocaracter).

```
1<0,0,C(s)>
2<0,0,C(i)>
3<0,0,C(r)>
4<0,0,C(_)>
5<4,2,C(d)>
6<4,1,C(e)>
7<0,0,C(a)>
8<6,1,C(t)>
9<0,0,C(m)>
10<4,1,C(n)>
11<8,4,C(i)>
12<0,0,C(l)>
13<0,0,C(y)>
14<7,1,C(t)>
15<8,3,C(e)>
16<2,1,C(_)>
17<4,2,C(a)>
18<4,2,C(i)>
19<0,0,C(c)>
20<0,0,C(k)>
21<9,4,C(l)>
```

Questão **2**

Correto

Atingiu 1,00 de 1,00

Caso se codifique trecho de poema de Cruz e Souza “vozes veladas veludosas vozes voluptuosas dos violões vozes veladas vagam nos velhos vórtices velozes dos ventos vivas as vulcanizadas” utilizando a mesma codificação LZ77 com dezesseis elementos do SB e no LAB, a saída teria

60

✓ símbolos, sendo

47

✓ símbolos diferentes (que não se repetem)

```
1<0,0,C(v)>
2<0,0,C(o)>
3<0,0,C(z)>
4<0,0,C(e)>
5<0,0,C(s)>
6<0,0,C( )>
7<6,1,C(e)>
8<0,0,C(l)>
9<0,0,C(a)>
10<0,0,C(d)>
11<2,1,C(s)>
12<8,4,C(u)>
13<8,1,C(o)>
14<8,1,C(a)>
15<10,3,C(o)>
16<24,5,C(o)>
17<16,2,C(p)>
18<0,0,C(i)>
19<15,3,C(d)>
20<21,2,C( )>
21<13,1,C(i)>
22<14,2,C(o)>
23<21,5,C(z)>
24<6,4,C(e)>
25<13,1,C(a)>
26<22,1,C(a)>
27<8,3,C(a)>
28<0,0,C(g)>
29<2,1,C(m)>
30<6,1,C(n)>
31<20,1,C(s)>
32<18,4,C(h)>
33<7,4,C(o)>
34<0,0,C(r)>
35<0,0,C(t)>
36<0,0,C(i)>
37<0,0,C(c)>
38<12,1,C(s)>
39<16,4,C(o)>
40<0,0,C(z)>
41<8,3,C(d)>
42<28,5,C(n)>
43<21,1,C(o)>
44<7,3,C(i)>
45<2,1,C(a)>
46<6,3,C(a)>
47<4,3,C(u)>
48<29,1,C(c)>
49<7,1,C(n)>
50<15,1,C(z)>
51<4,1,C(d)>
52<13,2,C($)>
```

Questão **3**

Correto

Atingiu 1,00 de 1,00

A **imagem2** é lida em escala de cinza. Se a imagem for lida continuamente (sem blocos) e convertida em um vetor usando `reshape(1, comprimento total)` pode-se assumir que todos seus coeficientes serão valores de 0 a 255. Decidiu-se aplicar LZ77 com LAB e SB de 100 elementos para codificar os coeficientes desse vetor. Fazendo isso, verificou-se que o número total de símbolos codificados foi de

4049

✓ sendo

1983

✓ códigos diferentes.

O processo é o mesmo, apenas deve-se ter o cuidado de fazer a leitura da imagem e sua transformação em vetor linear.

Questão **4**

Completo

Vale 2,00 ponto(s).

Desenvolva uma função `[code, SB_out, LAB_out] = fLZ77(SB_in, LAB_in)`, onde entrando-se o Search Buffer (SB_in) e o buffer de Look Ahead (LAB_in) a função retorne o próximo código e os novos valores de SB e LAB (SB_out e LAB_out) após o deslocamento da janela.

Obs.: As entradas LAB_in e SB_in são vetores e seu comprimento irá definir o comprimento dos respectivos buffers. Desse modo, quando a função for usada lembre de inicializar as mesmas nos valores desejados de tamanho para cada uma delas. Cole o código na caixa de texto da resposta.

```
function [code,SB_out,LAB_out] = fLZ77(SB_in, LAB_in)
arguments
SB_in (1,😊 char
LAB_in (1,😊 char
end
if isempty(SB_in) || isempty(LAB_in)
return;
end
match = 0;
offset = 0;
%i desliza o offset
for i=0:length(SB_in)-1
%ignora os matches depois de achar o primeiro
if match ~= 0
continue;
end
%j desliza a comparacao das janelas
for j=1:min(length(SB_in)-i, length(LAB_in))
if SB_in(1+i:j+i) == LAB_in(1:j)
match = j;
offset = i;
end
end
end
if match == length(LAB_in)
match = match -1;
end
code = sprintf('%d,%d,%c', offset,match,LAB_in(match+1));
SB_out = [SB_in LAB_in(1:match+1)];
LAB_out = LAB_in(match+2:length(LAB_in));
end
```

Questão **5**

Correto

Atingiu 1,00 de 1,00

A frase 'sir_sid_eastman_easily_teases_sea_sick_seals' é codificada com LZ 78. Em aula foram apresentados os primeiros passos do processo de codificação. Se a mensagem segue sendo codificada os códigos no alfabeto para os últimos quatro passos de codificação seriam

Passo 20:

(20,_si

✓ Passo 21:

(21,c)

✓ Passo 22

(22,k)

✓ Passo 23

(23,_se)

✓ Passo 24:

(24,al)

✓ . Coloque as respostas no formato (indice,novocaracter).

{ 's' }
 { 'si' }
 { 'r' }
 { '_' }
 { 'sid' }
 { '_e' }
 { 'a' }
 { 'st' }
 { 'm' }
 { 'an' }
 { '_ea' }
 { 'sil' }
 { 'y' }
 { '_t' }
 { 'e' }
 { 'as' }
 { 'es' }
 { '_s' }
 { 'ea' }
 { '_si' }
 { 'c' }
 { 'k' }
 { '_se' }
 { 'al' }

Questão 6

Correto

Atingiu 1,00 de 1,00

Caso se codifique trecho de poema de Cruz e Souza “vozes veladas veludasas vozes volupias dos violoes vozes veladas vagam nos velhos vortices velozes dos ventos vivas vas vulcanizadas” utilizando a mesma codificação LZ78, a saída teria

69

✓ símbolos, sendo

47

✓ símbolos diferentes (que não se repetem)

{'v' }
{'vo' }
{'z' }
{'e' }
{'s' }
{' ' }
{'ve' }
{'l' }
{'a' }
{'d' }
{'as' }
{' v' }
{'el' }
{'u' }
{'do' }
{'sa' }
{'s ' }
{'voz' }
{'es' }
{' vo' }
{'lu' }
{'p' }
{'i' }
{'as ' }
{'dos' }
{' vi' }
{'o' }
{'lo' }
{'es ' }
{'voze' }
{'s v' }
{'ela' }
{'da' }
{'s va' }
{'g' }
{'am' }
{' n' }
{'os' }
{' ve' }
{'lh' }
{'os ' }
{'vor' }
{'t' }
{'ic' }
{'es v' }
{'elo' }
{'ze' }
{'s d' }

{ 'os v' }
{ 'en' }
{ 'to' }
{ 's vi' }
{ 'va' }
{ 's vas' }
{ 'vu' }
{ 'lc' }
{ 'an' }
{ 'iz' }
{ 'ad' }

Questão **7**

Correto

Atingiu 1,00 de 1,00

A [imagem2](#) é lida em escala de cinza. Se a imagem for lida continuamente (sem blocos) e convertida em um vetor usando reshape(1, comprimento total) pode-se assumir que todos seus coeficientes serão valores de 0 a 255. Decidiu-se aplicar LZ8 para codificar os coeficientes desse vetor. Fazendo isso, verificou-se que o número total de símbolos codificados foi de

2954

✓ sendo

2789

✓ códigos diferentes.

O processo é o mesmo, apenas deve-se ter o cuidado de fazer a leitura da imagem e sua transformação em vetor linear.

Questão **8**

Completo

Vale 2,00 ponto(s).

Desenvolva uma função que codifica LZ78 uma dada string faz saída = codificaLZ78 (string de entrada) onde o valor de saída é um vetor de N linhas que cada linha indica um código de saída do codebook na forma [i,s] onde i é o índice do símbolo no dicionário e s é o símbolo. Cole o código na caixa de texto da resposta.

```
% codificacao LZ78 implementada de forma recursiva
% retorna o codebook, onde i eh um vetor de indices e s eh um vetor de char
% a entrada eh uma string
function [i,s] = codificaLZ78(input)
input = string(input);
i(1) = 0; s(1) = input{1};
bufferStart = 2;
bufferEnd = 2;
nodeIndexToInsert = 0;
while bufferStart < strlen(input)
buffer = input{1}(bufferStart:bufferEnd);
bufferNodeIndexToInsert = getNodeIndexToInsert(i, s, buffer);
if bufferNodeIndexToInsert ~= 0
nodeIndexToInsert = bufferNodeIndexToInsert;
end
if bufferNodeIndexToInsert ~= 0 && bufferEnd < strlen(input)
bufferEnd = bufferEnd+1;
continue
end
newIndex = length(i) + 1;
i(newIndex) = nodeIndexToInsert;
s(newIndex) = input{1}(bufferEnd);
bufferStart = bufferEnd + 1;
bufferEnd = bufferStart;
nodeIndexToInsert = 0;
end
end
% recebe o codebook e um no, retorna a string de forma recursiva
function expression = getNodeStringValue(i, s, index, char)
if(index == 0)
expression = char;
else
expression = [getNodeStringValue(i, s, i(index), s(index)) char];
end
end
% procura alguma expressao no codebook que tenha match com o buffer
% começa com os maiores indices, retorna 0 se não encontrar
function index = getNodeIndexToInsert(i, s, buffer)
index = 0;
for j=length(i):-1:1
nodeString = getNodeStringValue(i, s, i(j), s(j));
if strcmp(nodeString, buffer)
index = j;
return;
end
end
end
```

[◀ Texto Complementar 2](#)

Seguir para...

Tarefa 5 - Métodos de Dicionário ►