בע"ה

**דחיסה**

מספר הקורס: 89-745-01

סמסטר א מועד א

תאריך: 21.1.13 ; י שבט תשע"ג

מרצה: פרופ' שמואל קליין

משך הבחינה: שעתיים

*הנחיות*: א) אין להשתמש בכל חומר עזר, גם לא מחשבונים

ב) ענה/י על כל השאלות

ג) כל תשובה חייבת להיות מנומקת היטב

[1] (40 נקודות)

נתון הטקסט הבא, שאותיותיו לקוחות מתוך האלף-בית {a, b, c, d, e, f, g, h}

b d e c c g e a b d e c c c c f g a h d e f e a b d e c f d d e a b c c c c c f

התדירויות של האותיות הן:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| h | g | f | e | d | c | b | a |
| 1 | 2 | 4 | 7 | 6 | 12 | 4 | 4 |

1. בנה קוד הופמן 3-ary עבור אלף בית זה לפי תדירות הופעת האותיות בטקסט. השתמש בתוים A, B, C, כדי לציין את מילות הקוד על בסיס 3-ary , ואח"כ החלף כל אחד מהתוים במחרוזת בינארית באורך 1 או 2 כדי למזער את גודל הקובץ הדחוס.

One of the possible codes is

AA

AB

AC

B

CA

CB

CCA

CCC

That is, 6 As, 3 Bs and 7 Cs. So a possible coding is A=10, B=11, C=0

Which gives lengths

AA 4

AB 4

AC 3

B 2

CA 3

CB 3

CCA 4

CCC 3

And the total length will be 4 \*(6+4+1) + 3\*(7+4+4+2) + 12\*2 = 44+51+24=119 bits.

ב) חשב את אחוז הדחיסה שהתקבל בסעיף א) יחסית להצפנת הטקסט בקוד שבו כל אותי מילות הקוד שווים.

Using fixed length for 8 elements 🡪 3 bits per character, 40 chars = 120 bits.

So only 1 of 120 bits saved.

ג) הצפן את הטקסט לפי שיטת Lempel-Ziv-Welch ובנה במקביל את הטבלה.

Encoding:

2 4 5 3 3 7 5 1 --- 9 11 12 3 6 7 1 8 10 6 13 17 20 4 10 16 --- 19 12 6

Table:

1 a 11 ec 21 fg 31 dea

2 b 12 cc 22 ga 32 abc

3 c 13 cg 23 ah 33 cccc

4 d 14 ge 24 hd 34 ccf

5 e 15 ea 25 def

6 f 16 ab 26 fe

7 g 17 bde 27 eab

8 h 18 ecc 28 bdec

9 bd 19 ccc 29 cfd

10 de 20 ef 30 dd

ד) מהו אחוז הדחיסה המתקבל (יחסית להצפנה ב fixed-length-code) אם המצביעים לטבלה מוצפנים.

1. ב fixed-length.

Table has 34 entries, need 6 bits to encode index, 27 elements,

So 6 \* 27 = 162 bits 🡪 negative compression vs fixed length per char

2. בצורה הדרגתית, כאשר מתחילים עם טבלה בגודל מינמלי ומכפילים את גודלה כאשר זה נחוץ?

First 8 elements 4 bits, next 16 with 5 bits, last 3 with 6 bits, together 130 bits, better, but still worse than without compression.

[**2**] (40 נקודות)

נתונה קבוצה גדולה של מפות של סיביות, כל מפה בעלת אותו אורך N שהוא מספר גדול (למשל 50000) הבעיה שהן צורכות הרבה שטח, אך לרוב הן דלילות מאוד (מספר אחדים קטן מאוד לעומת מספר אפסים גדול). ראינו כמה שיטות לדחוס מפות כאלה, אך כל השיטות דוחסות כל מפה לחוד. ניתן אולי לשפר זאת אם דוחסים את הקבוצה כולה וניתן להשתמש במפה אחת כדי לדחוס אחרת.

(10) א) הראה/י שאם משתמשים בקידוד אריתמטי, מידת הדחיסה של מפה (compressibility) תלויה בדלילותה.

For arithmetic coding, the average size is the entropy, which, in the binary case is

-p log p – (1-p) log (1-p). For p<0.5 this is an increasing function, so if the maps are sparse, that is, p is close to 0, then the smaller p, the sparser the map and also the smaller is the compressed file.

(10) ב) אם מפה A ומפה B דומות מאוד, אז המפה C=A xor B תהיה דלילה יותר מ-A וגם מ-B. איך נוכל לנצל זאת?

Instead of keeping A and B, we keep A and C, and we can recover B by Xoring again, since B = A xor C.

(20) ג) איך נפעיל את הרעיון ב-ב) על כל קבוצת המפות? הדרכה: צריך למזער את מספר הביטים הדלוקים במפות הרגילות ובאלה שנשמרים בעזרת xor.

Consider the set of all maps and add to it the zero map Z. Build a full graph: the nodes are the maps (including Z), and set the weight of edge (A,B) to be the Hamming distance (number of different bits) from A to B. In fact, this is the number of 1-bits in A xor B. Build a minimum spanning tree T, locate Z in T and orient all edges towards Z. The maps pointing directly to Z are those kept on their own. Any other map A points to a map B, and instead of A we keep A xor B.

[**3**] (20 נקודות)

קוד פיבונצ'י הוא סדרה אינסופית של מילות קוד

{11, 011, 0011, 1011, 00011, 10011, 01011, 000011, 100011, ...}

שבה מילות הקוד מאופיינות ע"י כך שכולן מסתיימות ב- 11 וכן שהמחרוזת 11 אינה מופיעה באף מקום אחר באף מילת קוד. בדומה לכך אפשר להגדיר קוד טריבונצ'י:

{111, 0111, 00111, 10111, 000111, 100111, 010111, 110111, 0000111, 1000111, 0100111, 1100111, 0010111, 1010111, 0110111, 00000111, 10000111, …}

שבה מילות הקוד מאופיינות ע"י כך שכולן מסתיימות ב- 111 וכן שהמחרוזת 111 אינה מופיעה באף מקום אחר באף מילת קוד. סדרת טריבונצ'י מוגדרת ע"י

T(0)=T(1)=1, T(2)=2. T(i)= T(i-1) + T(i-2) + T(i-3) for i>2.

1. מה הקשר בין סדרת טריבונצ'י לקוד טריבונצ'י?

The numbers of codewords of every length are tribonacci numbers

1. דון/י ביתרונות וחסרונות של קוד טריבונצ'י לעומת קוד פיבונצ'י.

Advantage: more tribonacci codewords for a given length

Disadvantage: longer shortest codewords.

Other properties remain the same (robustness, fixed set).

בהצלחה !!!