Cấu trúc dữ liệu và giải thuật

NÉN DỮ LIỆU

Giảng viên:
Văn Chí Nam – Nguyễn Thị Hồng Nhung – Đặng Nguyễn Đức Tiến

Nội dung trình bày

- Giới thiệu
- Một số khái niệm
- Giải thuật nén Huffman tĩnh

Giới thiệu

3

- Thuật ngữ:
  - Data compression
  - Encoding
  - Decoding
  - Lossless data compression
  - Lossy data compression
  - …

Giới thiệu

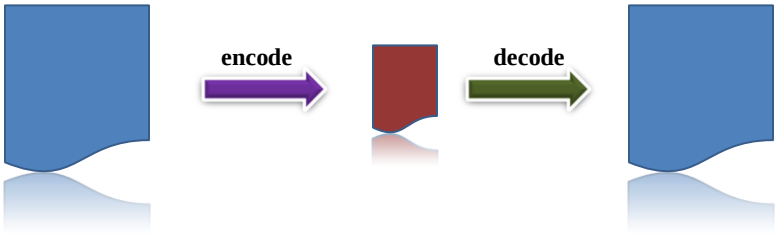
4

- Nén dữ liệu
  - Nhu cầu xuất hiện ngay sau khi hệ thống máy tính đầu tiên ra đời.
  - Hiện nay, phục vụ cho các dạng dữ liệu đa phương tiện
  - Tăng tính bảo mật.
- Ứng dụng:
  - Lưu trữ
  - Truyền dữ liệu

Giới thiệu

5

- Nguyên tắc:
  - Encode và decode sử dụng cùng một scheme.



Khái niệm

6

- Tỷ lệ nén (Data compression ratio)
  - Tỷ lệ giữa kích thước của dữ liệu nguyên thủy và của dữ liệu sau khi áp dụng thuật toán nén.
  - Gọi:
    - N là kích thước của dữ liệu nguyên thủy,
    - N_1 là kích thước của dữ liệu sau khi nén.
    - Tỷ lệ nén R: $$R = \frac{N}{N_1}$$
  - Ví dụ:
    - Dữ liệu ban đầu 8KB, nén còn 2 KB. Tỷ lệ nén: 4-1

Khái niệm

7

- Tỷ lệ nén (Data compression ratio)
  - Về khả năng tiết kiệm không gian: Tỷ lệ của việc giảm kích thước dữ liệu sau khi áp dụng thuật toán nén.
  - Gọi:
    - N là kích thước của dữ liệu nguyên thủy,
    - N_1 là kích thước của dữ liệu sau khi nén.
    - Tỷ lệ nén R: $$R = 1 - \frac{N_1}{N}$$
  - Ví dụ:
    - Dữ liệu ban đầu 8KB, nén còn 2 KB. Tỷ lệ nén: 75%

Khái niệm

8

- Nén dữ liệu không mất mát thông tin (Lossless data compression)
  - Cho phép dữ liệu nén được phục hồi nguyên vẹn như dữ liệu nguyên thủy (lúc chưa được nén).
  - Ví dụ:
    - Run-length encoding
    - LZW
    - …
  - Ứng dụng:
    - Ảnh PCX, GIF, PNG,..
    - Tập tin *. ZIP
    - Ứng dụng gzip (Unix)

Khái niệm

9

- Nén dữ liệu mất mát thông tin (Lossy data compression)
  - Dữ liệu nén được phục hồi
    - không giống hoàn toàn với dữ liệu nguyên thủy;
    - gần đủ giống để có thể sử dụng được.
  - Ứng dụng:
    - Dùng để nén dữ liệu đa phương tiện (hình ảnh, âm thanh, video):
      - Ảnh: JPEG, DjVu;
      - Âm thanh: AAC, MP2, MP3;
      - Video: MPEG-2, MPEG-4

10

Nén Huffman tĩnh

Giới thiệu

11

- Mong muốn:
  - Một giải thuật nén bảo toàn thông tin;
  - Không phụ thuộc vào tính chất của dữ liệu;
  - Ứng dụng rộng rãi trên bất kỳ dữ liệu nào, với hiệu suất tốt.

Giới thiệu

12

- Tư tưởng chính:
  - Phương pháp cũ: dùng 1 dãy bit cố định để biểu diễn 1 ký tự
  - David Huffman (1952): tìm ra phương pháp xác định mã tối ưu trên dữ liệu tĩnh :
    - Sử dụng vài bit để biểu diễn 1 ký tự (gọi là "mã bit" – bit code)
    - Độ dài "mã bit" cho các ký tự không giống nhau:
    - Ký tự xuất hiện nhiều lần: biểu diễn bằng mã ngắn;
    - Ký tự xuất hiện ít : biểu diễn bằng mã dài

  => Mã hóa bằng mã có độ dài thay đổi (Variable Length Encoding)

Giới thiệu

13

- Giả sử có dữ liệu sau đây:

  ADDAABBCCBAAABBCCCBBBCDAADDEEAA

Ký tự	Tần số xuất hiện
A	10
B	8
C	6
D	5
E	2

- Biểu diễn 8 bit/ký tự cần:

  (10 + 8 + 6 + 5 + 2) * 8 = **248 bit**

Giới thiệu

14

- Dữ liệu:

  ADDAABBCCBAAABBCCCBBBCDAADDEEAA

- Biểu diễn bằng chiều dài thay đổi:

Ký tự	Tần số	Mã
A	10	11
B	8	10
C	6	00
D	5	011
E	2	010

(10*2 + 8*2 + 6*2 + 5*3 + 2*3) = **69 bit**

Thuật toán nén

15

[B1]: Duyệt tập tin -> Lập bảng thống kê tần số xuất hiện của các ký tự.

[B2]: Xây dựng cây Huffman dựa vào bảng thống kê tần số xuất hiện

[B3]: Phát sinh bảng mã bit cho từng ký tự tương ứng

[B4]: Duyệt tập tin -> Thay thế các ký tự trong tập tin bằng mã bit tương ứng.

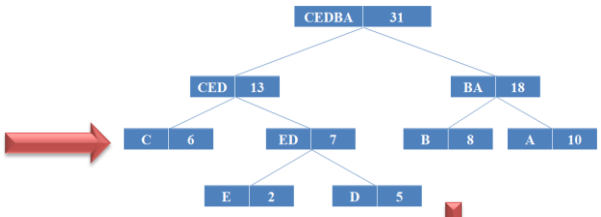
[B5]: Lưu lại thông tin của cây Huffman cho giải nén

Thuật toán nén

16

ADDAABBCCBAAABBCCCBBBCDAADDEEAA

Ký tự	Tần số
A	10
B	8
C	6
D	5
E	2



Ký tự	Mã
A	11
B	10
C	00
D	011
E	010

11011011111110100000101111111010000
00010101000011111101101101001011111

Thuật toán nén – Thống kê tần số

17

- Dữ liệu:

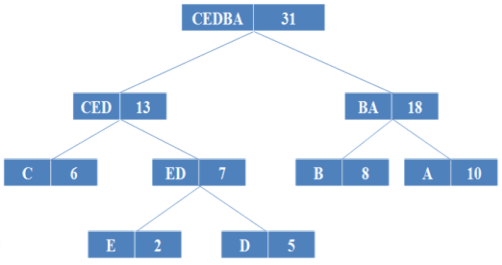
ADDAABBCCBAAABBCCCBBBCDAADDEEAA

Ký tự	Tần số xuất hiện
A	10
B	8
C	6
D	5
E	2

Thuật toán nén – Tạo cây Huffman

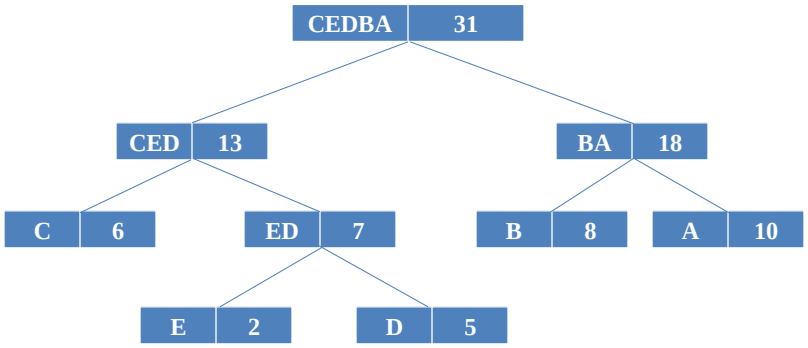
18

- Cây Huffman: cây nhị phân
  - Mỗi node lá chứa 1 ký tự
  - Mỗi node cha chứa các ký tự của những node con.
  - Trọng số của node:
    - Node con: tần số xuất hiện của ký tự tương ứng
    - Node cha: Tổng trọng số của các node con.



Thuật toán nén – Tạo cây Huffman

19



Thuật toán nén – Tạo cây Huffman

20

- Phát sinh cây:
  - Bước 1: Chọn trong bảng thống kê hai phần tử ***x,y*** có trọng số thấp nhất.
  - Bước 2: Tạo 2 node của cây cùng với node cha ***z*** có trọng số bằng tổng trọng số của hai node con.
  - Bước 3: Loại 2 phần tử ***x,y*** ra khỏi bảng thống kê.
  - Bước 4: Thêm phần tử ***z*** vào trong bảng thống kê.
  - Bước 5: Lặp lại Bước 1-4 cho đến khi còn 1 phần tử trong bảng thống kê.

Thuật toán nén – Tạo cây Huffman

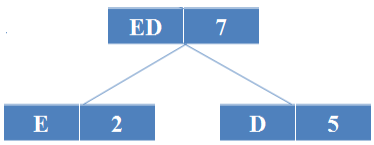
21

- Quy ước:
  - Node có trọng số nhỏ hơn sẽ nằm bên nhánh trái. Node còn lại nằm bên nhánh phải.
  - Nếu 2 node có trọng số bằng nhau
    - Node nào có ký tự nhỏ hơn thì nằm bên trái
    - Node có ký tự lớn hơn nằm bên phải.

Thuật toán nén – Tạo cây Huffman

22

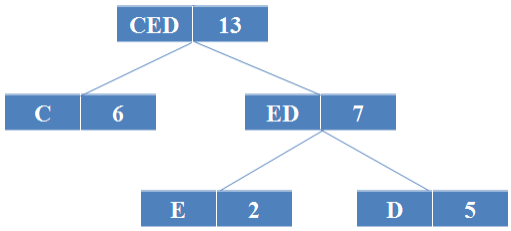
Ký tự	Tần số
A	10
B	8
C	6
D	5
E	2



Thuật toán nén – Tạo cây Huffman

23

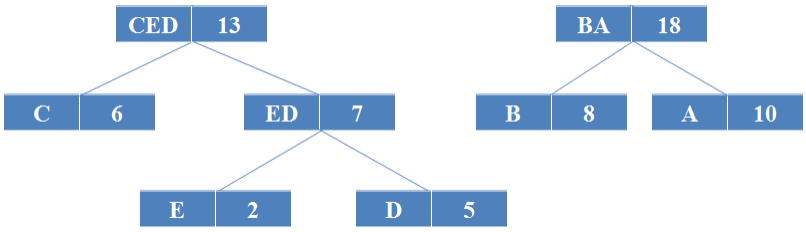
Ký tự	Tần số
A	10
B	8
ED	7
C	6



Thuật toán nén – Tạo cây Huffman

24

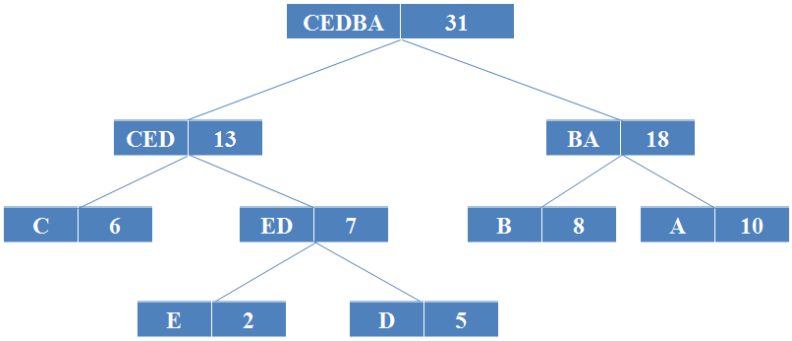
Ký tự	Tần số
CED	13
A	10
B	8



Thuật toán nén – Tạo cây Huffman

25

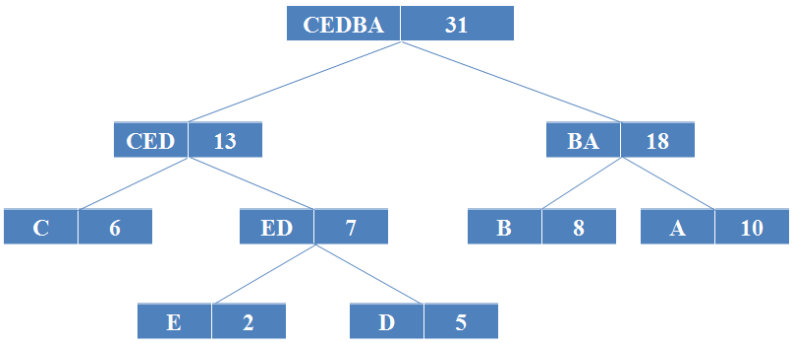
Ký tự	Tần số
BA	18
CED	13



Thuật toán nén – Tạo cây Huffman

26

Ký tự	Tần số
CEDBA	31



Thuật toán nén – Phát sinh mã bit

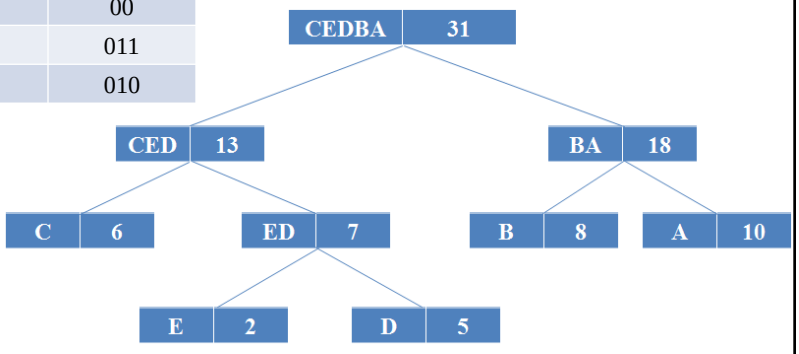
27

- Mã bit của từng ký tự: đường đi từ node gốc của cây Huffman đến node lá của ký tự đó.
- Cách thức:
  - Bit 0 được tạo ra khi đi qua nhánh trái
  - Bit 1 được tạo ra khi đi qua nhánh phải

Thuật toán nén – Phát sinh mã bit

28

Ký tự	Mã
A	11
B	10
C	00
D	011
E	010



Thuật toán nén – Nén dữ liệu

29

- Duyệt tập tin cần nén
- Thay thế tất cả các ký tự trong tập tin bằng mã bit tương ứng của nó.

Thuật toán nén – Lưu lại thông tin

30

- Phục vụ cho việc giải nén.
- Cách thức:
  - Cây Huffman
  - Bảng tần số

Thuật toán giải nén

31

- Phục hồi cây Huffman dựa trên thông tin đã lưu trữ.
- Lặp
  - Đi từ gốc cây Huffman
  - Đọc từng bit từ tập tin đã được nén
    - Nếu bit 0: đi qua nhánh trái
    - Nếu bit 1: đi qua nhánh phải
    - Nếu đến node lá: xuất ra ký tự tại node lá này.
- Cho đến khi nào hết dữ liệu

Vấn đề khác

32

- Có thể không lưu trữ cây Huffman hoặc bảng thống kê tần số vào trong tập tin nén hay không?

Vấn đề khác

33

- Thống kê sẵn trên dữ liệu lớn và tính toán sẵn cây Huffman cho bộ mã hóa và bộ giải mã.
- Ưu điểm:
  - Giảm thiểu kích thước của tập tin cần nén.
  - Giảm thiểu chi phí của việc duyệt tập tin để lập bảng thống kê
- Khuyết điểm:
  - Hiệu quả không cao trong trường hợp khác dạng dữ liệu đã thống kê

34

Hỏi và Đáp