Họ và tên: Trương Xuân Dũng – B18DCCN103

Lớp: E18CN01 – nhóm 10

STT: 05

**Báo cáo HW#3-02**

Kết quả trả về file **ketquamahoa.txt** và trên màn hình console.

Ví dụ này cũng thử nhập từ file **input.doc** khi nhập dữ liệu từ hàm **data\_file().**

**Thiết kế:**

Xây dựng class **node** chứa các thông tin cần thiết cho việc mã hóa **Huffman** như sau:

|  |
| --- |
| **Class: node** |
| **val(float):** Xác suất của nút  **left(node\*):** Con trỏ trỏ tới phần tử trái  **right(node\*):** Con trỏ trỏ tới phần tử phải  **l(char):** giá trị nhị phân trên đoạn nối với phần tử trái  **r(char):** giá trị nhị phân trên đoạn nối với phần tử phải  **isleaf(bool):** giá trị Boolean trả về nếu nút là nút lá  **data(char):** nếu đây là nút là thì lưu giá trị của chữ cái cần mã hóa |
| **node(float val):** hàm khởi tạo một nút với giá trị xác suất val |

**Các bước thực hiện và cài đặt:**

B1: Sử dụng **map** trong C++ để thực hiện thao tác **map** từng chữ cái trong xâu s với 1 giá trị xác suất tương ứng, sau đó push từng cặp **pair<char,float\*>** vào một **vector** là **vector < pair<char,float\*> > data** để xử lý tiếp.

B2: Tạo một vector lưu các con trỏ tới nút là **vector <node\*> save**, sau đó push từng giá trị của **data** vào đó để xây dựng các nút lá trước tiên.

B3: Ta nhận thấy rằng số các nút tối đa có thể có trong cây là **2\*data.size()-1**, nên dễ dàng kiểm soát các nút:

* Chạy vòng lặp *while* trong **save** sao cho trong mỗi vòng, tìm được 2 nút **save[vị trí 1]** và **save[vị trí 2]** có 2 giá trị xác suất **val** bé nhất mà chưa được đánh dấu.
* Sau đó tạo 1 nút **tmp** mới với:
  + giá trị xác suất mới **val = save[pos1]->val + save[pos2]->val**;
  + **left** trỏ tới **save[pos1]** với **l** = ‘0’;
  + **right** trỏ tới **save[pos2]** với **r** = ‘1’;
* sau đó push **tmp** vào **save**, đánh dấu 2 nút kia lại để tránh dùng lặp lại vòng tiếp theo (**flag** = false).
* *while* kết thúc khi **save.size() = 2\*data.size()-1;**

l = 0

l = 0

l = 0

r = 1

r = 1

r = 1

r = 1

l = 0

B4: Sau khi hoàn thiện cây, lưu gốc trên cùng là **root**. Ta duyệt từ **root** ngược lại như sau:

Nếu **root** trở thành lá, in ra xâu mã đã hình thành từ việc duyệt (**s\_here**), đồng thời tính **ntb** vì đã biết được độ dài mã tương ứng với kí tự.

Nếu không, duyệt trái hoặc phải nếu bên đó không mang giá trị **NULL**.

l = 0

l = 0

l = 0

r = 1

r = 1

r = 1

r = 1

l = 0

1

01

001

0001

0000

B5: Tính **Entropy H(X)**, đồng thời ta có luôn **ntb** đã tính.