**Bài thực hành**

**Bài thực hành 10: Cài đặt bài tập tìm chuỗi chung dài nhất**

Bài toán có thể được phát biểu như sau: viết chương trình nhập vào hai chuỗi kí tự, in ra độ dài chuỗi chung lớn nhất và chuỗi chung đó của hai chuỗi vừa nhập.

**Giải**

Truy vết để tìm và in ra xâu con ra màn hình sẽ dựa vào bảng kết quả mà ta tìm được ở phần trên. Bắt đầu từ vị trí L[n][m] và dừng lại khi L[i][j] == 0.

Bắt đầu duyệt từ i = n , j = m. Phần tử thứ i của xâu a sẽ là a[i-1], thứ j của xâu b sẽ là b[j-1]

Nếu a[i-1] == b[j-1] ta sẽ lưu lại con chung a[i-1] và giảm i và j đi 1 đơn vị

Nếu a[i-1] != b[j-1] có 2 trường hợp:

– Nếu L[i-1][j] >= L[i][j-1] thì giảm i 1 đơn vị

– Ngược lại giảm j đi 1 đơn vị

**Code**

#include<iostream>

#include<string>

using namespace std;

void longest(string a, string b){ // Hàm tìm chuỗi chung dài nhất và in ra màn hình

int n = a.size(); // n chiều dài chuỗi a, m chiều chuỗi xâu b

int m = b.size();

int max\_Size; // Biến lưu độ dài con chung lớn nhất

string temp = ""; // Biến lưu con chung dùng khi truy vết

int L[n+1][m+1]; // Khai báo mảng lưu kết quả: n+1 hàng, m+1 cột

for(int i=0; i<=n; i++) // Gán cột đầu tiên bằng 0

L[i][0] = 0;

for(int j=0; j<=m; j++) // Gán hàng đầu tiên = 0

L[0][j] = 0;

for(int i = 1; i<=n; i++){

for(int j = 1; j<=m; j++){

if(a[i-1] == b[j-1]){ // Nếu có phần tử bằng nhau

L[i][j] = L[i-1][j-1] + 1; // Áp dụng công thức

}

else{ // Trường hợp a[i-1] khác b[j-1]

if(L[i-1][j] >= L[i][j-1]) // Tìm max giữa L[i-1][j] và L[i][j-1]

L[i][j] = L[i-1][j];

else

L[i][j] = L[i][j-1];

}

}

}

max\_Size = L[n][m]; // Tìm được độ dài chuỗi chung lớn nhất

int i = n;

int j = m;

while(L[i][j] != 0){ // Điều kiện dừng

if(a[i-1] == b[j-1]){ // Nếu bằng nhau

temp += a[i-1]; // Cộng a[i-1] vào chuỗi chung

i--;

j--;

}

else{ // Nếu khác nhau

if(L[i-1][j] >= L[i][j-1]) // So sánh

i--;

else

j--;

}

}

cout<<"\nDo dai xau lon nhat: "<<max\_Size; // In ra độ dài chuỗi chung lớn nhất

cout<<"\nXau con: ";

for(int t = max\_Size-1 ; t>=0; t--) // In ngược từ cuối về đầu để chuỗi chung đúng thứ tự

cout<<temp[t];

}

int main(){

string a, b;

cout<<"Nhap chuoi a: ";

cin>>a;

cout<<"Nhap chuoi b: ";

cin>>b;

longest(a,b);

return 0;

}

**Kết quả**

