Double Hash for string:

```cpp
struct DoubleHash{
    const int p1=1333, p2=137; // 31 , 53 , 137 , 1331 , 1333
    const int m1=1e9+9, m2=1e9+7; // 1e9+7 , 1e9+9
    vector<ll> pow1, pow2, h1, h2;

    void build(string s) {
        int n = s.size();

        pow1.resize(n);
        h1.resize(n);
        pow1[0]=1;
        for(int i=1 ; i<n ; i++) pow1[i] = pow1[i-1]*p1%m1;
        h1[0]=s[0]-'a'+1;
        for(int i=1 ; i<n ; i++) h1[i] = (h1[i-1]*p1 + s[i]-'a'+1)%m1;

        pow2.resize(n);
        h2.resize(n);
        pow2[0]=1;
        for(int i=1 ; i<n ; i++) pow2[i] = pow2[i-1]*p2%m2;
        h2[0]=s[0]-'a'+1;
        for(int i=1 ; i<n ; i++) h2[i] = (h2[i-1]*p2 + s[i]-'a'+1)%m2;
    }

    void add(string s){
        for (int i = 0; i < s.size(); i++) pow1.push_back(pow1.back()*p1%m1);
        for (int i = 0; i < s.size(); i++) h1.push_back((h1.back()*p1 + s[i]-'0'+1)%m1);

        for (int i = 0; i < s.size(); i++) pow2.push_back(pow2.back()*p2%m2);
        for (int i = 0; i < s.size(); i++) h2.push_back((h2.back()*p2 + s[i]-'0'+1)%m2);
    }

    pair<int, int> getHash(int i, int j){
        ll a = h1[j];
        if(i) a -= h1[i-1] * pow1[j-i+1];
        a = (a%m1 + m1)%m1;

        ll b = h2[j];
        if(i) b -= h2[i-1] * pow2[j-i+1];
        b = (b%m2 + m2)%m2;

        return make_pair(a,b);
    }
};
```