CNC ASSIGNMENT

**SUBMITTED BY : SALONI SETH**

**17BCS045**

**BTECH CSE**

**SEMESTER 4**

#include <iostream>

#include <string>

#define MINUS\_ONE -1

const int ARRYMAX( 4\*100000 + 5);

using namespace std;

int TSites, Result, Count, FilterLen, Len;

char Filters[ARRYMAX], Fil[ARRYMAX];

struct TRIE

{

TRIE\* Child[27];

char Ch;

bool EOS;

bool Blocked;

TRIE(): Ch('\n'), EOS(false), Blocked(true) { for(int i = 0; i<27 ; ++i) { Child[i] = NULL; } }

TRIE(const char ch, const string& Website, const int Index, const bool IsBlocked ): Ch(ch), EOS(Website.size() == Index), Blocked(IsBlocked)

{

if(Index < Website.size())

{

const char NextCh = Website.at(Index);

const int NextI = NextCh - 'a';

int i = 0;

for(; i<NextI ; ++i)

{

Child[i] = NULL;

}

Child[i] = new TRIE(NextCh, Website, Index + 1, IsBlocked);

++i;

for(; i<27 ; ++i)

{

Child[i] = NULL;

}

}

else

{

for(int i=0; i<27 ; ++i)

{

Child[i] = NULL;

}

}

}

void Clean()

{

TRIE \*Pnt;

for(int i = 0; i<27 ; ++i)

{

Pnt = Child[i];

if(Pnt != NULL)

{

Child[i] = NULL;

delete Pnt;

}

}

}

~TRIE()

{

Clean();

}

void Insert(const string& Website, const int Index, const bool IsBlocked)

{

if( (EOS && Blocked && !IsBlocked) || (!Blocked && IsBlocked && Website.size() == Index))

{

Result = MINUS\_ONE;

return;

}

if(Blocked && !IsBlocked)

{

Blocked = false;

}

if(Website.size() > Index)

{

const char ch = Website.at(Index);

const int i = ch - 'a';

if(Child[i] == NULL)

{

Child[i] = new TRIE(ch, Website, Index + 1, IsBlocked);

}

else

{

Child[i]->Insert(Website, Index + 1, IsBlocked);

}

}

}

void FindFilters()

{

Fil[++Len] = Ch;

int i;

if(Blocked)

{

++Count;

for(i = 0; i<= Len; ++i)

{

Filters[++FilterLen] = Fil[i];

}

}

else

{

for( i = 0; i<27; ++i)

{

if(Child[i] != NULL)

Child[i]->FindFilters();

}

}

--Len;

}

};

int main ()

{

int TestCases, j, k;

char Oper;

string Website;

TRIE Trie;

cin >> TSites;

Result = Count = 0;

FilterLen = Len = -1;

for(j=0; j<TSites; ++j)

{

cin >> Oper >> Website;

if(Result != -1)

{

if(Oper == '-')

{

Trie.Insert(Website, 0, true);

}

else

{

Trie.Insert(Website, 0, false);

}

}

}

if(Result == -1)

{

cout << "-1";

}

else

{

Trie.FindFilters();

Filters[++FilterLen] = '\0';

cout << Count << Filters;

}

Trie.Clean();

}