**字符串包含**

**题目**

给定两个分别由字母组成的字符串A和字符串B，字符串B的长度比字符串A短。请问，如何最快地判断字符串B中所有字母是否都在字符串A里？

为了简单起见，我们规定输入的字符串只包含大写英文字母，请实现函数bool StringContains(string &A, string &B)

**算法1**

先对字符串A和B的字符进行排序

依次比较A和B字符串包含的字符

如果A[i]<B[i]

i++

如果i>=A.length或者A[i]>B[j]

A不包含B

j++

如果j>=B.length,A包含B

**代码1**

**package** stringContain;

**import** java.util.Arrays;

**import** java.util.Scanner;

**public** **class** StringContain {

**private** **static** **boolean** stringContain(String str1,String str2) {

**char**[] a=str1.toCharArray();

**char**[] b=str2.toCharArray();

Arrays.*sort*(a);

Arrays.*sort*(b);

**for**(**int** i=0,j=0;j<b.length;) {

**while**((i<a.length)&&(a[i]<b[j])){

i++;

}

**if**(i>=a.length||a[i]>b[j]) {

**return** **false**;

}

j++;

}

**return** **true**;

}

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

Scanner sc = **new** Scanner(System.***in***);

String str1=sc.nextLine();

String str2=sc.nextLine();

System.***out***.println(*stringContain*(str1,str2));

}

}

**算法2**

使用素数来标识字母A、B、C…、Z

遍历输入的字符A,把A中字符转换成对应的素数，统计它们的乘积

遍历输入的字符B,同样统计它们的乘积

用字符串A的乘积%（取余）字符串B的乘积，如果结果为0，A包含B。否则不包含

**代码2**

**private** **static** **boolean** stringContain1(String str1,String str2) {

**char**[] a=str1.toCharArray();

**char**[] b=str2.toCharArray();

**int** [] c= {2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59,61, 67, 71, 73, 79, 83, 89, 97, 101};

**int** f=1;

**int** g=1;

**for**(**int** i=0;i<a.length;i++) {

f=f\*(c[a[i]-'A']);

}

**for**(**int** j=0;j<b.length;j++) {

g=g\*(c[b[j]-'A']);

}

//System.out.println(f%g);

**if**(f%g==0)

**return** **true**;

**return** **false**;

}

**复杂度**

O(m+n)

**算法3**

1. 把字符串A的所有字符存储到HashTable
2. 判断B中的字符是不是都可以在哈希表中找到，如果不能，则A不包含B，否则A不包含B

**代码3**

**package** stringContain;

**import** java.util.Arrays;

**import** java.util.Hashtable;

**import** java.util.Scanner;

**public** **class** StringContain {

**private** **static** **boolean** stringContain(String str1,String str2) {

**char**[] a=str1.toCharArray();

**char**[] b=str2.toCharArray();

Arrays.*sort*(a);

Arrays.*sort*(b);

**for**(**int** i=0,j=0;j<b.length;) {

**while**((i<a.length)&&(a[i]<b[j])){

i++;

}

**if**(i>=a.length||a[i]>b[j]) {

**return** **false**;

}

j++;

}

**return** **true**;

}

**private** **static** **boolean** stringContain1(String str1,String str2) {

**char**[] a=str1.toCharArray();

**char**[] b=str2.toCharArray();

**int** [] c= {2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59,61, 67, 71, 73, 79, 83, 89, 97, 101};

**int** f=1;

**int** g=1;

**for**(**int** i=0;i<a.length;i++) {

f=f\*(c[a[i]-'A']);

}

**for**(**int** j=0;j<b.length;j++) {

g=g\*(c[b[j]-'A']);

}

//System.out.println(f%g);

**if**(f%g==0)

**return** **true**;

**return** **false**;

}

**private** **static** **boolean** stringContain2(String str1,String str2) {

**char**[] a=str1.toCharArray();

**char**[] b=str2.toCharArray();

Hashtable<Character , Character> ht=**new** Hashtable<Character, Character>();

**for**(**int** i=0;i<a.length;i++) {

ht.put(a[i], a[i]);

}

**for**(**int** i=0;i<b.length;i++) {

**if**(!ht.containsKey(b[i]))

**return** **false**;

}

**return** **true**;

}

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

Scanner sc = **new** Scanner(System.***in***);

String str1=sc.nextLine();

String str2=sc.nextLine();

System.***out***.println(*stringContain2*(str1,str2));

}

}