

第六章 字符数组与字符串

第 6 9 课 认识 `string`

《信息学奥赛一本通·编程启蒙 C++版》

二、string 的几种常用语法

限于篇幅，下面罗列了 string 的几种常用的语法，读者如果想要获得更多的关于 string 的用法，可以打开网页：<https://cplusplus.com>，然后搜索 string。

语法	说明
=	赋以新值
+=	在尾部添加字符
+	串联字符串
==, !=, <, <=, >, >=	比较字符串
[]	存取单一字符
>>, getline()	从 stream 读取某值
<<	将某值写入 stream
s. substr(i, l)	返回从索引 i 开始，长度为 l 的某个子字符串
s. find(x)	返回符合搜索条件的字符区间内的第一个字符的索引，没找到目标就返回 npos。
s. erase(i, l);	从索引 i 开始往后删 l 个
s. insert(i, str)	在 string 中间的某个位置插入字符串，指定一个安插位置的索引，被插入的字符串将放在这个索引的后面。

【例 69.1】 过滤多余的空格

【题目描述】

一个句子中也许有多个连续空格，过滤掉多余的空格，只留下一个空格。

【输入格式】

一行，一个字符串（长度不超过 200），句子的头和尾都没有空格。

【输出格式】

过滤之后的句子。

【样例输入】

```
Hello      world.This is      c language.
```

【样例输出】

```
Hello world.This is c language.
```

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. string s;
4. int main(){
5.     cin>>s;
6.     cout<<s;
7.     while(cin>>s){
8.         cout<<" "<<s;
9.     }
10.    return 0;
11.}
```

【例 69.2】 矩阵交换行

【题目描述】

给定一个 5×5 的矩阵（数学上，一个 $r \times c$ 的矩阵是一个由 r 行 c 列元素排列成的矩形阵列），将第 n 行和第 m 行交换，输出交换后的结果。

【输入格式】

输入共 6 行，前 5 行为矩阵的每一行元素（元素为一个 0 到 9 之间的整数），元素与元素之间以一个空格分开。

第 6 行包含两个整数 m 、 n ，以一个空格分开。（ $1 \leq m, n \leq 5$ ）。

【输出格式】

输出交换之后的矩阵，矩阵的每一行元素占一行，元素之间以一个空格分开。

【样例输入】

```
1 2 2 1 2
5 6 7 8 3
9 3 0 5 3
7 2 1 4 6
3 0 8 2 4
1 5
```

【样例输出】

```
3 0 8 2 4
5 6 7 8 3
9 3 0 5 3
7 2 1 4 6
1 2 2 1 2
```

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. string s[6];
4. int x,y;
5. int main(){
6.     for(int i=1;i<=5;i++)
7.         getline(cin,s[i]);
8.     cin>>x>>y;
9.     swap(s[x],s[y]);
10.    for(int i=1;i<=5;i++)
11.        cout<<s[i]<<endl;
12.    return 0;
13.}
```

【例 69.3】 字符串中的空格

【题目描述】

现在有 n 个字符串，其中包含一些空格，现在希望你将所有字符串中的空格删除后输出。

【输入格式】

输入第一行只有一个正整数 n ，表示共有 n 个字符串。 $1 \leq n \leq 10000$ 。

第二行至第 $n+1$ 行共 n 行，每行是一个字符串(由大小写英文字母和空格组成)。

【输出格式】

输出有 $n+1$ 行，第一行只有一个正整数，表示总共删除的空格数。第二行至第 $n+1$ 行共 n 行，每行表示一个删除空格后的字符串（按照输入顺序）。

【样例输入】

```
3
Yan ke lei
iiii
acac aca c ac
```

【样例输出】

```
9
Yankelei
iiii
acacacacac
```


【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. string s,a[10005],x;
4. int cnt,n,len;
5. int main(){
6.     scanf("%d\n",&n);//cin>>n;getchar();
7.     for(int i=1;i<=n;i++){
8.         getline(cin,s);
9.         len=s.size();
10.        x="";
11.        for(int j=0;j<len;j++){
12.            if(s[j]==' ') cnt++;
13.            else x=x+s[j];
14.        }
15.        a[i]=x;
16.    }
17.    cout<<cnt<<endl;
18.    for(int i=1;i<=n;i++)
19.        cout<<a[i]<<endl;
20.    return 0;
21. }
```


【例 69.4】 回文串

【题目描述】

现在有一个字符串。请问这个字符串是否可以在恰好更改一个字符的情况，变成一个回文串？

【输入格式】

第一行为一个正整数 t ($1 < t < 100$)，表示测试点的询问个数。

往下 t 行都有一个长度不大于 10000 的字符串，字符串内只包含小写英文字母。

【输出格式】

对于每个询问，输出一行，如果可以，则输出 YES，否则输出 NO。

【样例输入】

```
3
abccaa
abbcca
abcda
```

【样例输出】

```
YES
NO
YES
```

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int main(){
4.     int t;
5.     cin>>t;
6.     for(int i=1;i<=t;i++){
7.         string s;cin>>s;
8.         int c=0,l=s.size( );
9.         for(int i=0,j=l-1;i<j;i++,j--)
10.            if(s[i]!=s[j])
11.                c++;
12.         if(c<=1)cout<<"YES"<<endl;
13.         else cout<<"NO"<<endl;
14.     }
15.     return 0;
16.}
```

练 69.1 单词的长度

【题目描述】

输入一行单词序列，相邻单词之间由 1 个或多个空格间隔，请对应地计算各个单词的长度。

注意，如果有标点符号（如连字符，逗号，句号），标点符号算作与之相连的词的一部分。没有被空格间隔开的符号串，都算作单词。

【输入格式】

一行单词序列，最少 1 个单词，最多 300 个单词，单词之间用至少 1 个空格间隔。单词序列总长度不超过 1000。

【输出格式】

依次输出对应单词的长度，之间以逗号间隔。

【样例输入】

She was born in 1990-01-02 and from Beijing city.

【样例输出】

3,3,4,2,10,3,4,7,5

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. string s;
4. int main(){
5.     cin>>s;
6.     cout<<s.size();
7.     while(cin>>s)
8.         cout<<','<<s.size();
9.     return 0;
10.}
```

练 69.2 单词倒排

【题目描述】

编写程序，读入一行英文(只包含字母和空格，单词间以单个空格分隔)，将所有单词的顺序倒排并输出，依然以单个空格分隔。

【输入格式】

输入为一个字符串（字符串长度至多为 **100**）。

【输出格式】

输出为按要求排序后的字符串。

【样例输入】

I am a student

【样例输出】

student a am I

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. string s[105];
4. int cnt;
5. int main(){
6.     while(cin>>s[++cnt]);
7.     cnt--;
8.     while(cnt){
9.         cout<<s[cnt]<<' ';
10.        cnt--;
11.    }
12.    return 0;
13.}
```

练 69.3 整理药名

【题目描述】

医生在书写药品名的时候经常不注意大小写，格式比较混乱。现要求你写一个程序将医生书写混乱的药品名整理成统一规范的格式，即药品名的第一个字符如果是字母要大写，其他字母小写。如将"ASPIRIN"、"aspirin"整理成"Aspirin"。

【输入格式】

第一行一个数字 n ，表示有 n 个药品名要整理， n 不超过 100。

接下来 n 行，每行一个单词，长度不超过 20，表示医生手书的药品名。
药品名由字母、数字和-组成。

【输出格式】

n 行，每行一个单词，对应输入的药品名的规范写法。

【样例输入】

```
4
AspiRin
cisapride
2-PENICILLIN
Cefradine-6
```

【样例输出】

```
Aspirin
Cisapride
2-penicillin
Cefradine-6
```


【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int n;
4. string s;
5. int main(){
6.     cin>>n;
7.     while(n--){
8.         cin>>s;
9.         if(islower(s[0])) s[0]=toupper(s[0]);
10.        for(int i=1,len=s.size();i<len;i++){
11.            if(isupper(s[i])) s[i]=tolower(s[i]);
12.        }
13.        cout<<s<<endl;
14.    }
15.    return 0;
16.}
```

练 69.4 信息加密

【题目描述】

在传递信息的过程中，为了加密，有时需要按一定规则将文本转换成密文发送出去。有一种加密规则是这样的：

将字母转换成其后的第 3 个字母，例如：A→D，a→d；对于字母表的最后 3 个字母，将其相应转化为字母表前 3 个字母，例如 X→A，x→a。

对于非字母字符，保持不变。

现在，请你根据输入的一行字符，输出其对应的密码。

【输入格式】

输入一个字符串。字符串长度不超过 100

【输出格式】

输出加密后的字符串。

【样例输入】

I(2016)love(08)China(15)!

【样例输出】

L(2016)oryh(08)Fklqd(15)!

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. char c;
4. int main(){
5.     while(cin>>c){
6.         if(c>='a'&&c<='w' || c>='A'&&c<='W') cout<<char(c+3);
7.         else if(c>='x'&&c<='z') cout<<char('a'+c-'x');
8.         else if(c>='X'&&c<='Z') cout<<char('A'+c-'X');
9.         else cout<<c;
10.    }
11.    return 0;
12.}
```

练 69.5 恺撒加密术(趣味编程)

【题目描述】

加密术最早应用于古代战争。古罗马时期，凯撒大帝曾是使用密码来传递信息，它是一种替代密码，对于信中的每个字母，会用它后面第 i 给字母代替。试编一个程序，将输入的一段英文字符加密后输出，只加密字母，加密的规则是用字母后来的第 1 个字母代替用来的字母，即用 'b' 代替 'a'，用 'c' 代替 'b'，..... 用 'a' 代替 'z'。

【输入格式】

一行字符串

【输出格式】

加密后的字符串。

【样例输入】

Nike is excellent!

【样例输出】

Ojlf jt fydfmmfou!

【代码实现】

```
1. #include<iostream>
2. #include<cstdio>
3. #include<cstring>
4. using namespace std;
5. string str1,str2;
6. int main(){
7.     char s;
8.     getline(cin,str1);
9.     int i;
10.    str2="";
11.    for (int i=0;i<str1.size();i++){
12.        s=str1[i];
13.        if ((s>='a'&&s<='z') || (s>='A'&&s<='Z')){//判断大小写
14.            s++; //加密
15.            if ((s>'Z' && s<'a') || s>'z'){
16.                s-=26;
17.            }
18.        }
19.        str2+=s;
20.    }
21.    cout<<str2;
22.    return 0;
23.}
```

谢谢！

